Learning from Heterogeneous Genomic Data

Habilitation Thesis

Jiří Kléma

August 31, 2012

Acknowledgements

I would like to thank all people who have helped and inspired me during the last decade. I owe a great deal of gratitude to my numerous colleagues and co-authors. In particular, I want to thank my closest and long-term CTU collaborators. Olga Štěpánková mentored my early scientific years in machine learning. Filip Železný has motivated me the most in recent bioinformatics years. I was delighted to collaborate with Bruno Cremilleux and his GREYC team. I am grateful to him for hiring me as an post-doc researcher and giving me a chance to consistently work on current genomic problems. Last but not least, I would like to express my deepest gratitude to my family for their love and support.

Contents

Preface

I am presenting this thesis to qualify for habilitation at the Czech Technical University in Prague. The thesis is a review of my recent bioinformatics research. Bioinformatics and namely the field of learning from heterogeneous genomic data became my primary research interest during my one year post-doc stay at the GREYC laboratory, University of Caen, France. In the years 2005-2006 I worked on the French national project on genomics and inductive databases "Bases de donnes INductives et GnOmique (BINGO)" and focused on mining plausible patterns from gene expression data and information extraction from genomic texts. We succeeded to prove that the background knowledge can help in efficient extraction of interpretable patterns in the form of bi-sets of genes and biological conditions. After my return to the home university and the Intelligent data analysis (IDA) research group, I continued in this line of research. In terms of the bilateral Czech-French project "Heterogeneous Data Fusion for Genomic and Proteomic Knowledge Discovery", we finished drafts of two book chapters and further developed the idea of utilization of the sequential patterns to discover named entities such as gene mentions in biomedical texts.

In parallel, I started to specialize in another way of bioinformatics research that stemmed from my long term experience in classification and learning from examples. My main goal was to study how far the background knowledge available on gene and protein functions and relations can improve accuracy and interpretability of molecular classifiers. At the same time, I co-initiated cooperation with several Czech biological institutes and labs such as the Department of Genetic Ecotoxicology of Institute of Experimental Medicine, Academy of Sciences, the Department of Nephrology of the Institute for Clinical and Experimental Medicine, or the Institute of Biology and Medical Genetics, 2nd Medical Faculty, Charles University. These links help to bring my research findings into practice and increase the knowledge of informaticians in needs of the biological laboratories. Last but not least, I am one of the founders and lecturers of a new Bioinformatics course taught in the newly starting study programme in Biomedical engineering and informatics. To summarize, I did my best to contribute and assist to the effort of the head of the IDA group Filip Železný to establish our group as one of the recognized Czech

bioinformatics teams. I believe that this effort was successful as our team regularly publishes in recognized impacted journals, cooperates with Czech biological institutes and mediates the field and recent findings to our students.

The thesis is organized as an annotated collection of 12 papers assorted from among my publications since the year 2007. Three papers were published in impacted scientific journals (the impact factors of 4.9, 3.0 and 1.7), two other in peer-reviewed international journals, three papers made chapters in international books and the other four articles appeared in conference proceedings. As I preferred consistency in topics to completeness of my research profile, I did not include my earlier or parallel works on other than bioinformatics topics as well as the conference articles that reasonably overlap with their later journal extensions. The full list of my publications can be accessed at http://labe.felk.cvut.cz/~klema/publ.html.

The thesis starts with a self-contained overview of the remaining chapters containing the individual papers. The papers are split in four chapters. Chapter 2 summarizes the research on mining patterns from gene expression data. Chapter 3 gives an overview of set-level microarray classification. Chapter 4 concerns information extraction from genomic texts. Chapter 5 provides the book chapter that reviews the research on gene expression mining guided by background knowledge which is the general topic that straddles the previous ones discussed in the Chapters 2-4. At the same time, the chapter exemplifies one of my successful bioinformatics applications solved in cooperation with the team of the Department of Genetic Ecotoxicology from Czech Academy of Sciences.

Chapter 1

Habilitation Thesis Overview

High-throughput technologies like microarrays allow researchers to simultaneously monitor the expression of tens of thousands of genes. They represent a valuable resource allowing to better understand diseases on a molecular level, predict gene and protein functions and assemble or particularize gene regulatory networks. However, gene-expression data analysis represents a difficult task as the data usually show an inconveniently low ratio of samples (biological situations) against variables (genes). Datasets are often noisy and contain a great part of variables irrelevant in the context under consideration. Consequently, the analysis of gene-expression data shall be driven, focused or at least verified against genomic background knowledge. The term genomic background knowledge refers to any information that is not directly available in a gene-expression dataset but it is related to the genes or biological situations contained in this dataset. Basically, the information that annotates, groups or links the genes as well as situations under study, the information that helps to regularize the resulting models. Learning and knowledge discovery proceeds in a bootstrapping manner, the background knowledge is used to model the data while the models improve the existing knowledge. This thesis deals with two particular ways of learning from gene expression data driven by background knowledge. It also suggests several possibilities of automated extraction of structured genomic knowledge from free biomedical texts.

Mining Patterns from Gene Expression Data

Gene-expression data facilitate an insight into gene function and regulatory mechanisms. In this type of gene-expression data analysis, the key step is the detection of groups of co-expressed genes, i.e., the genes that manifest similar expression patterns. Clustering provides the most straightforward and traditional approach to obtain co-expressed genes. The resulting partitioning is human understandable, the

number of gene groups is controllable and the hierarchical approaches enable to model a general gene taxonomy. However, it is well-known that a typical group of genes shares an activation pattern only under specific experimental conditions. The same group of genes behaves almost independently under the other conditions. Moreover, a single gene may be co-expressed with very diverse gene groups under different conditions as it may have multiple biological functions. The global model is not expressive enough to capture the true relationship between genes and biological situations.

An alternative way is to employ local methods and search for patterns – sets of genes displaying a specific expression characteristic in a set of situations. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate patterns which can hardly be further exploited. A timely application of background knowledge available in literature databases, gene ontologies and other sources can help to focus on the most plausible patterns only. In this chapter I present the papers that propose, implement and test a flexible constraint-based framework that enables the effective mining and representation of meaningful over-expression patterns representing intrinsic associations among genes and biological situations. The framework can be simultaneously applied to a wide spectrum of genomic data. I and my co-authors also demonstrate that it allows to generate new biological hypotheses with clinical implications.

The proposed framework falls into a large family of bi-clustering algorithms that tackle the above-mentioned shortcomings of global clustering. It is unique in the extent and ease of application of external constraints derived from background knowledge. We specialized in binarized expression data that only state whether or not a gene is expressed in a given situation. This feature was motivated by Serial Analysis of Gene Expression (SAGE) data that are binary and represent an alternative to the more frequent microarrays that provide real-valued expressions. Although SAGE itself is currently being dominated by microarrays, I believe it is not a limitation as a microarray outcome can be binarized prior to analysis.

The results presented in this chapter were reached in close collaboration with my French project leader Bruno Cremilleux. Bruno Cremilleux is a full professor who specializes namely in pattern discovery and constraint satisfaction. Since 2011 he heads the Department of Computer Science at the University of Caen. The second closest collaborator was Arnaud Soulet, then a PhD student supervised by Bruno Cremilleux, the author of the tool Music. Currently, Arnaud is a teaching assistant at the University of Tours. The raw genomic data as well as the verification of biological validity of the generated models were provided by Olivier Gandrillon and his team of Centre de Gntique Molculaire et Cellulaire, University of Lyon, France. The application of quantitative association rules to genomic data was developed with my former PhD student Filip Karel who successfully finished his PhD

on ordinal association rule mining in 2009.

Specifically, the chapter consists of the following papers:

- Soulet, A., Klema, J., Cremilleux, B.: Efficient Mining Under Rich Constraints Derived from Various Datasets. In Dzeroski, S., Struyf, J. (eds.): Knowledge Discovery in Inductive Databases, Lecture Notes in Computer Science Volume 4747/2007, Springer Berlin / Heidelberg, pp. 223-239, 2007.

- Karel, F., Klema, J.: Quantitative Association Rule Mining in Genomics. In Berendt, B., Svatek, V. Zelezny, F. (eds.): Proc. of The ECML/PKDD Workshop On Prior Conceptual Knowledge in Machine Learning and Data Mining. University of Warsaw, Poland, pp. 53-64, 2007.

- Klema, J., Blachon, S., Soulet, A., Cremilleux, B., Gandrilon, O.: Constraint-Based Knowledge Discovery from SAGE Data. In Silico Biology, 8, 0014, 2008.

- Cremilleux, B., Soulet, A., Klema, J., Hebert, C., Gandrillon, O.: Discovering Knowledge from Local Patterns in SAGE Data. In Berka, P., Rauch, J., Zighed, D.A. (eds.): Data Mining and Medical Knowledge Management: Cases and Applications, IGI Global Inc., pp. 251-267, 2009.[1]

Set-level Microarray Classification

Molecular classification of biological samples based on their gene-expression profiles is a natural learning task with immediate practical uses. Since the early success stories published 15 years ago, there was a large number of studies with the main goal of predicting cancer or other diseases. However, the routine application of gene-expression classification is limited by frequent inaccuracies in the resulting classifiers and their incomprehensibility for physicians. Consequently, molecular classifiers based solely on gene expression in most cases cannot be considered useful decision-making tools or decision-supporting tools.

Similarly to the domain of pattern mining, recent efforts in the field of molecular classification aim to employ background knowledge. The idea is to extract features that correspond to functionally related gene sets instead of the individual genes, respectively the probesets whose expression is available in the original expression data. The new features are supposed to be more robust as they filter out noise, easier to interpret because they correspond to more general biological phenomenons and less overfit since their number is limited when compared to the vast amount of genes. Last but not least, the set-level features technically allow to merge biological samples measured on different platforms and even taken from different species, i.e., the biological samples with the original feature vectors that do not match (different length, different probesets, different genes). The issues to study are obvious: 1) which genes shall be grouped, 2) how to compute the set-level expressions, and 3) how to identify the most prospective gene set candidates before learning.

The papers presented in this chapter study and resolve the above-mentioned issues of set-level genomic classification. In summary, we carried out several extensive sets of experiments. We created both features made strictly by the pre-defined gene sets corresponding to the individual biological terms or processes and possibly more heterogeneous ”free” features based on gene functional clustering. We tested simple aggregation functions such as averaging as well as singular value decomposition and other advanced methods that can cope with concurrent gene activation and inhibition. Gene sets were ranked with the recent dedicated gene-set methods, their results were compared with the results reached by the classical feature selection methods such as information gain. We studied both the single-platform as the cross-platform and cross-species scenario.

The results can be summarized as follows. First of all, the functional gene sets proved to outperform the random ones, the functionally related gene groups definitely make predictive features. The single platform experiments suggest that set-level classifiers do not boost predictive accuracy, however, they do achieve competitive accuracy if learned with the right combination of components. In cross-platform design, the set-level classification enables to reach larger sample sets and thus it results in more accurate classifiers when single-platform samples are rare.

This is an encouraging conclusion regarding interpretability of the set-level classifiers. Moreover, the competitive overall performance means that there is a reasonable portion of domains where set-level classification can clearly be recommended. The set features constructed by now were also general and not dedicated to the particular tasks, feature extraction aiming at specific domains can still bring an accuracy improvement.

There is one more significant output related to this topic. The IDA research group developed a public web tool XGENE.ORG for cross-genome and cross-organism gene expression data analysis. The tool makes a great portion of the implemented methods available to the biological community and it is routinely used by our Czech scientific partners under our assistance.

The results presented in this chapter were reached in collaboration with other members of IDA group. Filip Zelezny is the head of the group, Matěj Holec is one of his PhD students who specializes in set-level genomic classification and Miloš Krejník is my PhD student who graduated with a diploma thesis in functional genomic clustering and then switched to the field of time series prediction. Jakub Tolar from University of Minnesota motivated the initial phases of the research and provided a biological feedback.

In particular, the chapter contains the following papers:

- Holec, M., Klema, J., Zelezny, F., Belohradsky, J., Tolar, J.: Cross-Genome Knowledge-Based Expression Data Fusion. International Conference on Bioinformatics, Computational Biology, Genomics and Chemoinformatics (BCBGC-09), 2009.

- Holec M., Klema J., Zelezny F., Tolar J.: Comparative Evaluation of Set-Level Techniques in Predictive Classification of Gene Expression Samples. BMC Bioinformatics, 13, Suppl. 10, S15, 2012.

- Krejnik, M., Klema J.: Empirical Evidence of the Applicability of Functional Clustering through Gene Expression Classification. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 9:3, pp. 788-798, 2012.

Information Extraction from Genomic Texts

The previous chapters employ the available structural genomic knowledge to improve the analysis of gene expression data. This chapter offers several methods to create it from collections of free biomedical texts, namely the research papers and their short summaries. The structural databases are typically carefully manually curated, consequently they provide a precise and reliable resource of background knowledge. However, concerning coverage and timeliness, a large amount of the biological information is only available in natural language in research publications, technical reports, websites and other text documents. A critical challenge is then to automatically extract relevant and useful knowledge dispersed in such text collections.

My early joint work with Thierry Charnois and Nicolas Durand aimed at development of structured representation of gene summaries. These summaries represented one of the most valuable information knowledge resources as they are concise, contain the most valuable pieces of gene information and each summary is unambiguously attached to a single gene. The agent part of binary biological interaction is known by default, the goal was to find the target parts of these relations. The extraction was based on a declarative grammar tailored to the specific jargon of gene summaries. The semi-automatically created grammar proved solid precision and recall regarding its simplicity.

The subsequent work with Marc Plantevit and the French team on discovery of named entities such as gene and protein names in full texts aimed at the development of a fully automated approach. It led to the application of sequential data mining to the problem of named entity recognition as a counterpart to the heavy deep parsing methods and statistical approaches such as hidden Markov models or support vector machines. We took benefit from synergic action of pattern and rule mining techniques. Sequential patterns can hit a large spectrum of potentially interesting phrases while sequential rules bring necessary precision as they need to meet a confidence threshold. In order to monitor a wider context of the found sequential patterns, we relaxed the word ordering and treated the surrounding words as word sets in terms of frequent pattern mining. This was the additional way to reduce the relatively high false positive rate of sequential pattern mining.

We tried to further modify the sequential approach to gene interaction extraction by sentence skeletonization with my PhD student Přemysl Vítovec. We kept the framework of sequential data mining and looked at the fact that natural language is mostly not sequential in another way. We resolved this disproportion in terms of preprocessing that sequentializes the original text. Each sentence is decomposed into structurally simpler sequences of words called skeletons.

This topic is covered by the following triplet of papers:

- Charnois, T., Durand, N., Klema, J.: Automated Information Extraction from Gene Summaries. Humbolt Universitat Berlin, Germany, pp. 4-15, 2006.
- Plantevit, M., Charnois, T., Klema, J., Rigotti, C., Cremilleux, B.: Combining Sequence and Itemset Mining to Discover Named Entities in Biomedical Texts: A New Type of Pattern. International Journal of Data Mining, Modelling and Management, Vol. 1, No. 2, pp. 119-148, 2009.
- Vitovec, P., Klema, J.: Gene Interaction Extraction from Biomedical Texts by Sentence Skeletonization. In Znalosti 2011: VSB-TUO Ostrava, pp. 230-242, 2011.

Applications and Reviews

This chapter shows an application of the set-level approach discussed in the Chapter 3 to the particular domain of respirable ambient air particulate matter. It represents a complex mixture consisting of toxic and carcinogenic chemicals with harmful effects on human health. Namely the small particles with the diameter less than 2.5 micrometers, which is around 100 times thinner than a human hair, can cause severe respiratory problems. The team of the Department of Genetic Ecotoxicology of Institute of Experimental Medicine, Academy of Sciences studied the changes in the whole genome expression profiles induced by extractable organic matter in human embryonic lung fibroblasts. My role was namely to identify significantly deregulated gene sets. We found significantly deregulated genes and biological processes in various Czech localities with different sources and extent of air pollution. A prominent role of activation of aryl hydrocarbon receptor-dependent gene expression was suggested.

At the very end, I enclose a book chapter that summarizes and reviews the earlier applications of genomic background knowledge in several tasks such as relational descriptive analysis, constraint-based knowledge discovery, feature selection and construction or quantitative association rule mining. The chapter is focused namely on the applications carried out in IDA research group, it takes over and logically connects the work published in our earlier papers. Although the chapter is partly redundant with respect to the previous papers of this thesis, I believe it can give a comprehensible and hopefully readable summary of my bioinformatics activities in between of 2005-09.

There is one application paper and one review book chapter here:

- Libalova, H., Uhlirova, K., Klema, J., Machala, M., Sram, R., Ciganek, M. and Topinka, J.: Global Gene Expression Changes in Human Embryonic Lung Fibroblasts Induced by Organic Extracts from Respirable Air Particles. Particle and Fibre Toxicology, 9:1, 2012.

- Klema, J., Zelezny, F., Trajkovski, I., Karel, F., Cremilleux, B., Tolar, J.: Gene Expression Mining Guided by Background Knowledge. In Berka, P., Rauch, J., Zighed, D.A. (eds.): Data Mining and Medical Knowledge Management: Cases and Applications, IGI Global Inc., pp. 268-292, 2009.[2]

[2]This chapter was posted by permission of the publisher. Copyright 2009, IGI Global, www.igi-global.com.

Chapter 2

Mining Patterns from Gene Expression Data

Efficient Mining Under Rich Constraints Derived from Various Datasets

Arnaud Soulet[1], Jiří Kléma[1,2], and Bruno Crémilleux[1]

[1] GREYC, Université de Caen
Campus Côte de Nacre
F-14032 Caen Cédex France
{Forename.Surname}@info.unicaen.fr
[2] Department of Cybernetics
Czech Technical University, Prague
klema@labe.felk.cvut.cz

Abstract. Mining patterns under many kinds of constraints is a key point to successfully get new knowledge. In this paper, we propose an efficient new algorithm MUSIC-DFS which soundly and completely mines patterns with various constraints from large data and takes into account external data represented by several heterogeneous datasets. Constraints are freely built of a large set of primitives and enable to link the information scattered in various knowledge sources. Efficiency is achieved thanks to a new closure operator providing an interval pruning strategy applied during the depth-first search of a pattern space. A transcriptomic case study shows the effectiveness and scalability of our approach. It also demonstrates a way to employ background knowledge, such as free texts or gene ontologies, in the discovery of meaningful patterns.

Keywords: constraint-based mining, transcriptomic data.

1 Introduction

In current scientific, industrial or business data mining applications, the critical need is not to generate data, but to derive knowledge from huge and heterogeneous datasets produced at high throughput. In order to explore and discover new highly valuable knowledge it is necessary to develop environments and tools able to put all this data together. This involves different challenges, like designing efficient tools to tackle a large amount of data and the discovery of patterns of a potential user's interest through several datasets. There are various ways to interconnect the heterogeneous data sources and to express the mutual relations among the entities they address. Constraints provide a focus on the most promising knowledge by reducing the number of extracted patterns to those of a potential interest given by the user. Furthermore, when constraints can be pushed deep inside the mining algorithm, performance is improved, making the mining task computationally feasible and resulting in a human-workable output.

This paper addresses the issue of efficient pattern mining from large binary data under flexible constraints derived from additional heterogeneous datasets

S. Džeroski and J. Struyf (Eds.): KDID 2006, LNCS 4747, pp. 223–239, 2007.

synthetizing background knowledge (BK). Large datasets are characterized mainly by a large number of columns (i.e., items). This characteristic often encountered in a lot of domains (e.g., bioinformatics, text mining) represents a remarkable challenge. Usual algorithms show difficulties in running on this kind of data due to the exponential search space growth with the number of items. Known level-wise algorithms commonly fail in mining frequent or constrained patterns in such data [17]. On top of that, the user often would like to integrate BK in the mining process in order to focus on the most plausible patterns consistent with pieces of existing knowledge. BK is available in relational and literature databases, ontological trees and other sources. Nevertheless, mining in a heterogeneous environment allowing a large set of descriptions at various levels of detail is highly non-trivial. This paper solves the problem by pushing user-defined constraints that may stem both from the mined binary data and the BK summarized in similarity matrices or textual files.

The contribution of this paper is twofold. First we provide a new algorithm Music-dfs which soundly and completely mines constrained patterns from large data while taking into account external data (i.e., several heterogeneous datasets). Except for specific constraints for which tricks like the transposition of data [14, 9] or the use of the extension [8] can be used, levelwise approaches cannot tackle large data due to the huge number of candidates. On the contrary, Music-dfs is based on a depth first search strategy. The key idea is to use a new closure operator enabling an efficient interval pruning for various constraints (see Section 3). In [5], the authors also benefit from intervals to prune the search space, but their approach is restricted to the conjunction of one monotone constraint and one anti-monotone constraint. The output of Music-dfs is an interval condensed representation: each pattern satisfying the given constraint appears once in the collection of intervals only. Second, we provide a generic framework to mine patterns with a large set of constraints based on several heterogeneous datasets like texts or similarity matrices. It is a way to take into account the BK. Section 4 depicts a transcriptomic case study. The biological demands require to mine the expression data with constraints concerning complex relations represented by free texts and gene ontologies. The discovered patterns are likely to encompass interesting and interpretable knowledge.

This paper differs from our work in [20] for a double reason. First, the framework is extended to external data. Second, Music-dfs is deeply different from the prototype used in [20]: Music-dfs integrates primitives to tackle external data and thanks to its strategy to prune the search space (new interval pruning based on prefix-free patterns, see Section 3), it is able to mine large data. Section 4 demonstrates the practical effectiveness of Music-dfs in a transcriptomic case study and shows that other prototypes (including the prototype presented in [20]) fail. To the best of our knowledge, there is no other constraint-based tool to efficiently discover patterns from large data under a broad set of constraints linking the information distributed in various knowledge sources.

This paper is organized as follows. Section 2 introduces our framework to mine patterns satisfying constraints defined over several kinds of datasets. In Section 3,

we present the theoretical essentials that underlie the efficiency of MUSIC-DFS and we provide its main features. Experiments showing the efficiency of MUSIC-DFS and the cross-fertilization between several sources of genomic information are given in Section 4.

2 Defining Constraints on Several Datasets

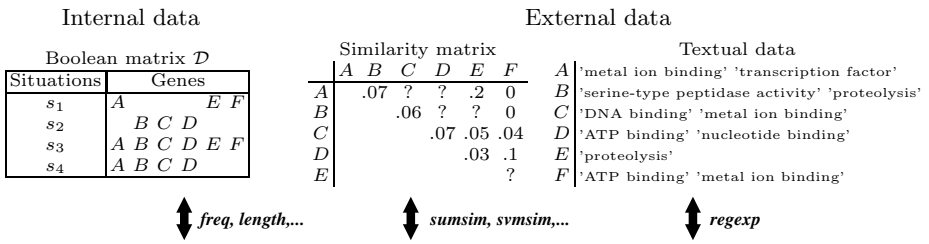
2.1 Integrating Background Knowledge Within Constraints

Usual data-mining tasks rarely deal with a single dataset. Often it is necessary to connect knowledge scattered in several heterogeneous sources. In constraint-based mining, the constraints should effectively link different datasets and knowledge types. In the domain of genomics, there is a natural need to derive constraints both from expression data and descriptions of the genes and/or biological situations under consideration. Such constraints require to tackle various data types - transcriptome data and background knowledge may be stored in the boolean, numeric, symbolic or textual format.

Let us consider the transcriptomic mining context given in Figure 1. Firstly, the involved data include a transcriptome dataset also called internal data. The dataset is in the transactional format - the items correspond to genes and the transactions represent biological situations. The occurrence of an item in a transaction signifies over-expression of the corresponding gene in the corresponding biological situation (genes A, E and F are over-expressed in situation s_1). Secondly, external data – a similarity matrix and textual resources – are considered. They summarize background knowledge that contains various information on items (i.e., genes). This knowledge is transformed into a similarity matrix and a set of texts. Each field of the triangular matrix $s_{ij} \in [0, 1]$ gives a similarity measure between the items i and j. The textual dataset provides a description of genes. Each row of this dataset contains a list of phrases characterizing the given gene (details are given in Section 4.1). The mined patterns are composed of items of the internal data, the corresponding transactions are usually also noted (and possibly analyzed). The external data are used to further specify constraints in order to focus on meaningful patterns. In other words, the constraints may stem from all the datasets.

Table 1 provides the meaning of the primitive constraints applied in this text. The meaning of the primitives is also illustrated by their real values taken from the example in Figure 1. As primitives can address different datasets, the dataset makes another parameter of the primitive (for clarity not shown in Table 1).

A real example of the compound constraint $q(X)$ is given in Figure 1. The first part (a) of q addresses the internal data and means that the biologist is interested in patterns having a satisfactory size – a *minimal area*. Indeed, $area(X) = freq(X) \times length(X)$ is the product of the frequency of X and its length and means that the pattern must cover a minimum number of situations and contain a minimum number of genes. The other parts deal with the external data: (b) is used to discard ribosomal patterns (one gene exception per pattern is allowed), (c) avoids



$$
\begin{aligned}
q(X) \equiv\ & freq(X) \times length(X) \geq 24 && \text{(a)}\\
& \wedge length(regexp(X, '*ribosom*', \texttt{TEXT\_terms})) \leq 1 && \text{(b)}\\
& \wedge svsim(X, \texttt{TEXT})/(svsim(X, \texttt{TEXT}) + mvsim(X, \texttt{TEXT})) \geq 0.7 && \text{(c)}\\
& \wedge sumsim(X, \texttt{TEXT})/svsim(X, \texttt{TEXT}) \geq 0.025 && \text{(d)}
\end{aligned}
$$

Fig. 1. Example of a toy (transcriptomic) mining context and a constraint

Table 1. Examples of primitives and their values in the data mining context of Figure 1. Let us note that item pairs of the pattern ABC are (A, B), (A, C) and (B, C).

Primitives		Values
Boolean matrix		
$freq(X)$	frequency of X	$freq(ABC) = 2$
$length(X)$	length of X	$length(ABC) = 3$
Textual data		
$regexp(X, RE)$	items of X whose associated phrases match the regular expression RE	$regexp(ABC, '*ion*') = AC$
Similarity matrix		
$sumsim(X)$	the similarity sum over the set of item pairs of X	$sumsim(ABC) = 0.13$
$svsim(X)$	the number of item pairs in X for which a similarity value is recorded	$svsim(ABC) = 2$
$mvsim(X)$	the number of item pairs in X for which a similarity value is missing	$mvsim(ABC) = 1$
$insim(X, min, max)$	the number of item pairs of X whose similarity lies between min and max	$insim(ABC, 0.07, 1) = 1$

patterns with prevailing items of an unknown function and (d) is to ensure a minimal average gene similarity. Section 4 provides another constraint q'.

Let us generalize the previous informal description. Let $\mathcal{I}$ be a set of items. A pattern is a non-empty subset of $\mathcal{I}$. $\mathcal{D}$ is a transactional dataset (or boolean matrix) composed of rows usually called transactions. A pattern X is present in $\mathcal{D}$ whenever it is included in one transaction of $\mathcal{D}$ at least. The constraint-based mining task aims to discover all the patterns present in $\mathcal{D}$ and satisfying a constraint q. Unfortunately, real constraints adressing several datasets (the constraint q, for example) are difficult to mine because they have no suitable property as monotonicity [12] or convertibility [16].

2.2 Primitive-Based Constraints

This section presents our framework previously defined in [20] (and the declarative language) enabling the user to set compound and meaningful constraints. This framework naturally integrates primitives adressing external data (e.g., *sumsim* or *regexp*). Furthermore, in our framework constraints are freely built of a large set of primitives. Beyond the primitives mentioned earlier there are primitives such as $\{\wedge, \vee, \neg, <, \leq, \subset, \subseteq, +, -, \times, /, sum, max, min, \cup, \cap, \setminus\}$. The compound constraints of this framework are called *primitive-based constraints*. There are no formal properties required on the final constraints. The only property which is required on the primitives to belong to our framework is a property of monotonicity according to each variable of a primitive (when the others remain constant) [20]. We have already shown that the whole set of primitive-based constraints constitutes a super-class of monotone, anti-monotone, succinct and convertible constraints [19]. Consequently, the proposed framework provides a flexible and rich constraint (query) language. The user can iteratively develop complex constraints integrating various knowledge types.

Let us recall that the primitives and the constraints defined in [20] only address one boolean data set. Current constraints can consider properties taken from a wide scale of dataset types. In addition to the similarity and textual datasets, the framework also enables to access numerical datasets having items in rows and numerical attributes in columns. It implements the primitive X.val which gives the list of values of the attribute named *val* for the items contained in the pattern X.

We give below other examples of constraints belonging to primitive-based constraints and highlighting the generality of our framework:

$$\begin{cases} freq(X) \times length(X) \geq 6 & \text{minimal area (nothing)} \\ (min(X.val) + max(X.val))/2 \leq 50 & \text{maximal mean (loose anti-monotone [2])} \\ sum(X.val)/length(X) \geq 25 & \text{minimal average (convertible [16])} \\ AE \subseteq X & \text{having } AE \text{ (monotone [12])} \\ freq(X) \geq 2 & \text{minimal frequency (anti-monotone [1])} \end{cases}$$

A previous work [21] approximates primitive-based constraints by one anti-monotone and one monotone constraint which can be pushed by DualMiner [5]. The next section describes an alternative solution in order to benefit from equivalence classes. This way is often more efficient because it avoids the enumeration of all the patterns which compose a particularly huge collection in the case of wide datasets. Besides, in context of wide datasets, previous algorithm Music [20] is ineffective due to the breadth-first search approach (see experiments in Section 4.2). Then, Section 3 presents a new algorithm dedicated to primitive-based constraints in wide datasets.

3 Music-dfs Tool

This section presents the Music-dfs tool (Mining with a User-Specifled Constraint, Depth-First Search approach) which benefits from the primitive-based

constraints presented in the previous section. Efficiency is achieved thanks to the exploitation of the primitive and constraint properties. We start by giving the key idea of the safe pruning process based on intervals.

3.1 Main Features of the Interval Pruning

The pruning process performed by MUSIC-DFS is based on the key idea to exploit properties of the monotonicity of the primitives (see Section 2) on the bounds of intervals to prune them. This new kind of pruning is called *interval pruning*. Given two patterns $X \subseteq Y$, the interval $[X, Y]$, also called sub-algebra or sub-lattice, corresponds to the set $\{Z \subseteq \mathcal{I} \mid X \subseteq Z \subseteq Y\}$. Figure 2 depicts an example with the interval $[AB, ABCD]$ and the values of the primitives $sumsim$ and $svsim$.

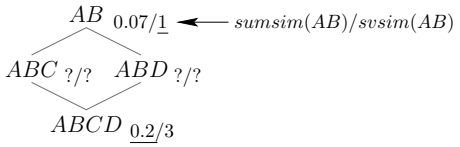


Fig. 2. Illustration of the interval pruning

Assume the constraint $sumsim(X)/svsim(X) \geq 0.25$. As the values associated to the similarities are positive, $sumsim(X)$ is an increasing function according X. Thus $sumsim(ABCD)$ is the highest $sumsim$ value for the patterns in $[AB, ABCD]$. Similarly, all the patterns of this interval have a higher $svsim(X)$ value than $svsim(AB)$. Thereby, each pattern in $[AB, ABCD]$ has its average similarity lower or equal than $sumsim(ABCD)/svsim(AB) = 0.2/1$. As this fraction does not exceed 0.25, no pattern of $[AB, ABCD]$ can satisfy the constraint and this interval can be pruned. We say that this pruning is *negative* because no pattern satisfies the constraint. In the same way, if the values of proper combinations of the primitives on the bounds of an interval $[X, Y]$ show that all the patterns in $[X, Y]$ satisfy the constraint, then $[X, Y]$ is also pruned and this pruning is named *positive*. For instance, assuming that $sumsim(AB)/svsim(ABCD) \geq 0.02$, then all the patterns in $[AB, ABCD]$ satisfy the constraint.

In a more formal way, this approach is performed by two interval pruning operators $\lfloor . \rfloor$ and $\lceil . \rceil$ introduced in [20] (but only for primitives handling boolean data). The main idea of these operators is to recursively decompose the constraint to benefit from the monotone properties of the primitives and then to safely negatively or positively prune intervals as depicted above. This process is straightforwardly extended to all the primitives, no matter what kind of dataset they regard. This highlights the generic properties of our framework, as well as the feature of pushing all the parts of the constraint q into the mining step. Table 2 gives the description of the lower and upper bounding operators corresponding to the previous examples of primitives. In Table 2, the general notation

Table 2. The definitions of $\lfloor . \rfloor$ and $\lceil . \rceil$ with particular primitives

$e \in \mathcal{E}_i$	Primitive(s)	$\lfloor e \rfloor \langle X, Y \rangle$	$\lceil e \rceil \langle X, Y \rangle$
$e_1 \theta e_2$	$\theta \in \{\wedge, \vee, +, \times, \cup, \cap\}$	$\lfloor e_1 \rfloor \langle X, Y \rangle \theta \lfloor e_2 \rfloor \langle X, Y \rangle$	$\lceil e_1 \rceil \langle X, Y \rangle \theta \lceil e_2 \rceil \langle X, Y \rangle$
$e_1 \theta e_2$	$\theta \in \{>, \geq, \supset, \supseteq, -, /, \setminus\}$	$\lfloor e_1 \rfloor \langle X, Y \rangle \theta \lceil e_2 \rceil \langle X, Y \rangle$	$\lceil e_1 \rceil \langle X, Y \rangle \theta \lfloor e_2 \rfloor \langle X, Y \rangle$
θe_1	$\theta \in \{\neg, freq, \}$	$\theta \lceil e_1 \rceil \langle X, Y \rangle$	$\theta \lfloor e_1 \rfloor \langle X, Y \rangle$
$\theta(e_1.val)$	$\theta \in \{min\}$	$\theta(\lceil e_1 \rceil \langle X, Y \rangle.val)$	$\theta(\lfloor e_1 \rfloor \langle X, Y \rangle.val)$
$\theta(e_1)$	$\theta \in \{length\}$	$\theta \lfloor e_1 \rfloor \langle X, Y \rangle$	$\theta \lceil e_1 \rceil \langle X, Y \rangle$
$\theta(e_1.val)$	$\theta \in \{sum, max\}$	$\theta(\lfloor e_1 \rfloor \langle X, Y \rangle.val)$	$\theta(\lceil e_1 \rceil \langle X, Y \rangle.val)$
$\theta(e_1)$	$\theta \in \{sumsim, svsim, mvsim\}$	$\theta(\lfloor e_1 \rfloor \langle X, Y \rangle)$	$\theta(\lceil e_1 \rceil \langle X, Y \rangle)$
$\theta(e_1, m, M)$	$\theta \in \{insim\}$	$\theta(\lfloor e_1 \rfloor \langle X, Y \rangle, m, M)$	$\theta(\lceil e_1 \rceil \langle X, Y \rangle, m, M)$
$\theta(e_1, RE)$	$\theta \in \{regexp\}$	$\theta(\lfloor e_1 \rfloor \langle X, Y \rangle, RE)$	$\theta(\lceil e_1 \rceil \langle X, Y \rangle, RE)$
$c \in E_i$	-	c	c
$X \in \mathcal{L}_\mathcal{I}$	-	X	Y

E_i designates one space among $\mathfrak{B}$, $\Re^+$ or $\mathcal{L}_\mathcal{I} = 2^\mathcal{I}$ and $\mathcal{E}_i$ the associated expressions (for instance, the set of constraints $\mathcal{Q}$ for the booleans $\mathfrak{B}$).

The next section indicates how the intervals are built.

3.2 Interval Condensed Representation

As indicated in Section 1, levelwise algorithms are not suitable to mine datasets with a large number of items due to the huge number of candidates growing exponentially according to the number of items. We adopt a depth-first search strategy instead of enumerating the candidate patterns and avoiding subsequent memory failures. We introduce a new and specific closure operator based on a prefix ordering relation $\preceq$. We show that this closure operator is central to the interval condensed representation (Theorem 1) and enables efficient pruning of the search space.

The prefix ordering relation $\preceq$ starts from an arbitrary order over items $A < B < C < \ldots$ as done in [16]. We say that an ordered pattern $X = x_1 x_2 \ldots x_n$ (i.e., $\forall i < j$, we have $x_i < x_j$) is a prefix of an ordered pattern $Y = y_1 y_2 \ldots y_m$ and note $X \preceq Y$ iff we have $n \leq m$ and $\forall i \in \{1, \ldots, n\}$, $x_i = y_i$. For instance, the prefixes of $ABCD$ are the patterns A, AB, ABC and $ABCD$. On the contrary, $AD \not\preceq ADC$ because the ordered form of ADC corresponds to ACD, and AD is not a prefix of ACD.

Definition 1 (Prefix-closure). *The prefix-closure of a pattern X, denoted $\mathbf{cl}_\preceq(X)$, is the pattern $\{a \in \mathcal{I} | \exists Y \subseteq X$ such that $Y \preceq Y \cup \{a\}$ and $freq(Ya) = freq(Y)\}$.*

The pattern $\mathbf{cl}_\preceq(X)$ gathers together the items occurring in all the transactions containing $Y \subseteq X$ such that Y is a prefix of $Y \cup \{a\}$. The fixed points of operator $\mathbf{cl}_\preceq$ are named the *prefix-closed patterns*. Let us illustrate this definition on our running example (cf. Figure 1). The pattern ABC is not a prefix-closed pattern

because ABC is a prefix of $ABCD$ and $freq(ABCD) = freq(ABC)$. On the contrary, $ABCD$ is prefix-closed. We straightforwardly deduce that any pattern and its prefix-closure have the same frequency. For instance, as $\mathbf{cl}_{\preceq}(ABC) = ABCD$, $freq(ABC) = freq(ABCD) = 2$.

A closure operator is a function satisfying three main properties: extensivity, isotony, and idempotency [22]. Next property shows that $\mathbf{cl}_{\preceq}$ is a closure operator:

Property 1 (Closure operator). *The prefix-closure operator* $\mathbf{cl}_{\preceq}$ *is a closure operator.*

Proof. *Extensivity:* Let X be a pattern and $a \in X$. We have $\{a\} \subseteq X$ and obviously, $a \preceq a$ and $freq(a) = freq(a)$. Then, we obtain that $a \in \mathbf{cl}_{\preceq}(X)$ and $\mathbf{cl}_{\preceq}$ is extensive. *Isotony:* Let $X \subseteq Y$ and $a \in \mathbf{cl}_{\preceq}(X)$. There exists $Z \subseteq X$ such that $Z \preceq Za$ and $freq(Za) = freq(Z)$. As we also have $Z \subseteq Y$ (and $freq(Za) = freq(Z)$), we obtain that $a \in \mathbf{cl}_{\preceq}(Y)$ and conclude that $\mathbf{cl}_{\preceq}(X) \subseteq \mathbf{cl}_{\preceq}(Y)$. *Idempotency:* Let X be a pattern. Let $a \in \mathbf{cl}_{\preceq}(\mathbf{cl}_{\preceq}(X))$. There exists $Z \subseteq \mathbf{cl}_{\preceq}(X)$ such that $freq(Za) = freq(Z)$ with $Z \preceq Za$. As $Z \subseteq \mathbf{cl}_{\preceq}(X)$, for all $a_i \in Z$, there is $Z_i \subseteq X$ such that $freq(Z_i a_i) = freq(Z_i)$ with $Z_i \preceq Z_i a_i$. We have $\bigcup_i Z_i \preceq \bigcup_i Z_i a$ and $freq(\bigcup_i Z_i) = freq(\bigcup_i Z_i a)$ (because $freq(\bigcup_i Z_i) = freq(Z)$). As the pattern $\bigcup_i Z_i \subseteq X$, a belongs to $\mathbf{cl}_{\preceq}(X)$ and then, $\mathbf{cl}_{\preceq}$ is idempotent. □

Property 1 is important because it enables to infer results requiring the properties of a closure operator. First, this new prefix-closure operator designs *equivalence classes* through the lattice of patterns. More precisely, two patterns X and Y are equivalent iff they have the same prefix-closure (i.e., $\mathbf{cl}_{\preceq}(X) = \mathbf{cl}_{\preceq}(Y)$). Of course, as $\mathbf{cl}_{\preceq}$ is idempotent, the maximal pattern (w.r.t. $\subseteq$) of a given equivalence class of X corresponds to the prefix-closed pattern $\mathbf{cl}_{\preceq}(X)$. Conversely, we call *prefix-free patterns* the minimal patterns (w.r.t. $\subseteq$) of equivalence classes. Second, closure properties enable to prove that the prefix-freeness is an anti-monotone constraint (see Property 2 in the next section).

Contrary to the equivalence classes defined by the Galois closure [4, 15], equivalence classes provided by $\mathbf{cl}_{\preceq}$ have a unique prefix-free pattern. This allows to prove that a pattern belongs to one interval only and provides the important result on the interval condensed representation (cf. Theorem 1). This result cannot be achieved without the new closure operator. Lemma 1 indicates that any equivalence class has a unique prefix-free pattern:

Lemma 1 (Prefix-freeness operator). *Let* X *be a pattern, there exists an unique minimal pattern (w.r.t.* $\subseteq$*), denoted* $\mathbf{fr}_{\preceq}(X)$*, in its equivalence class.*

Proof. Supposing that X and Y are two minimal patterns of the same equivalence class: we have $\mathbf{cl}_{\preceq}(X) = \mathbf{cl}_{\preceq}(Y)$. As X and Y are different, there exists $a \in X$ such that $a \notin Y$ and $a \leq min_{\preceq}\{b \in Y \backslash X\}$ (or we swap X and Y). As X is minimal, no pattern $Z \subseteq X \cap Y$ satisfies that $Z \preceq Za$ and $freq(Za) = freq(Z)$. Besides, for all Z such that $Y \cap X \subset Z \subset Y$, we have $Z \not\preceq Za$ because a is smaller than any item of $Y \backslash X$. So, a does not belong to $\mathbf{cl}_{\preceq}(Y)$ and then, we

obtain that $\mathbf{cl}_{\preceq}(X) \neq \mathbf{cl}_{\preceq}(Y)$. Thus, we conclude that any equivalence class exactly contains one prefix-free pattern. □

Lemma 1 means that the operator $\mathbf{fr}_{\preceq}$ links a pattern X to the minimal pattern of its equivalence class, i.e. $\mathbf{fr}_{\preceq}(X)$. X is prefix-free iff $\mathbf{fr}_{\preceq}(X) = X$. Any equivalence class corresponds to an interval delimited by one prefix-free pattern and its prefix-closed pattern (i.e., $[\mathbf{fr}_{\preceq}(X), \mathbf{cl}_{\preceq}(X)]$). For example, AB (resp. $ABCD$) is the prefix-free (resp. prefix-closed) pattern of the equivalence class $[AB, ABCD]$.

Now let us show that the whole collection of the intervals formed by all the prefix-free patterns and their prefix-closed patterns provides an *interval condensed representation* where each pattern X is present only once in the set of intervals.

Theorem 1 (Interval condensed representation). *Each pattern X present in the dataset is included in the interval $[\mathbf{fr}_{\preceq}(X), \mathbf{cl}_{\preceq}(X)]$. Besides, the number of these intervals is less than or equal to the number of patterns.*

Proof. Let X be a pattern and $R = \{[\mathbf{fr}_{\preceq}(X), \mathbf{cl}_{\preceq}(X)] | freq(X) \geq 1\}$. Lemma 1 proves that X is exactly contained in $[\mathbf{fr}_{\preceq}(X), \mathbf{cl}_{\preceq}(X)]$. The latter is unique. As X belongs to R by definition, we conclude that R is a representation of any pattern. Now, the extensivity and the idempotency of prefix-closure operator $\mathbf{cl}_{\preceq}$ ensure that $|R| \leq |\{X \subseteq \mathcal{I} \text{ such that } freq(X) \geq 1\}|$. This proves Theorem 1. □

In the worst case the size of the condensed representation is the number of patterns (each pattern is its own prefix-free and its own prefix-closed pattern). But, in practice, the number of intervals is low compared to the number of patterns (in our running example, only 23 intervals sum up the 63 present patterns).

The condensed representation highlighted by Theorem 1 differs from the condensed representations of frequent patterns based on the Galois closure [4, 15]: in this last case, intervals are described by a free (or key) pattern and its Galois closure and a frequent pattern may appear in several intervals. We claim that the presence of a pattern in a single interval brings meaningful advantages: the mining is more efficient because each pattern is tested at most once. This property improves the synthesis of the output of the mining process and facilitates its analysis by the end-user. The next section shows that by combining this condensed representation and the interval pruning operators, we get an interval condensed representation of primitive-based constrained patterns.

3.3 Mining Primitive-Based Constraints in Large Datasets

When running, MUSIC-DFS enumerates all the intervals sorted in a lexicographic order and checks whether they can be pruned as proposed in Section 3.1. The enumeration benefits from the anti-monotonicity property of the prefix-freeness (cf. Property 2). The memory requirements grow only linearly with the number of items and the number of transactions.

Property 2. *The prefix-freeness is an anti-monotone constraint (w.r.t. $\subseteq$).*

The proof of Property 2 is very similar to those of the usual freeness [4, 15]:

Proof. Let X be a pattern which is not a prefix-free pattern. So, there is $Z \subset X$ such that $\mathbf{cl}_{\preceq}(Z) = \mathbf{cl}_{\preceq}(X)$. Let Y be a pattern with $X \subseteq Y$. First, we observe that $\mathbf{cl}_{\preceq}(Y) = \mathbf{cl}_{\preceq}(X \cup (Y \setminus X))$ and $\mathbf{cl}_{\preceq}(X \cup (Y \setminus X)) = \mathbf{cl}_{\preceq}(\mathbf{cl}_{\preceq}(X) \cup \mathbf{cl}_{\preceq}(Y \setminus X))$ (usual property of closure operators). As $\mathbf{cl}_{\preceq}(Z) = \mathbf{cl}_{\preceq}(X)$, we obtain that $\mathbf{cl}_{\preceq}(\mathbf{cl}_{\preceq}(X) \cup \mathbf{cl}_{\preceq}(Y \setminus X)) = \mathbf{cl}_{\preceq}(\mathbf{cl}_{\preceq}(Z) \cup \mathbf{cl}_{\preceq}(Y \setminus X))$ and then, $\mathbf{cl}_{\preceq}(\mathbf{cl}_{\preceq}(Z) \cup \mathbf{cl}_{\preceq}(Y \setminus X)) = \mathbf{cl}_{\preceq}(Z \cup (Y \setminus X))$. Finally, as Z is a proper subset of X, the pattern $Z \cup (Y \setminus X)$ is a proper subset of Y. Thus, we conclude that Y is not prefix-free. □

In other words, the anti-monotonicity ensures us that once we know that a pattern is not prefix-free, any superset of this pattern is not prefix-free anymore [1, 12]. Algorithms 1 and 2 give the sketch of Music-dfs.

Algorithm 1. GlobalScan

Input: A prefix-pattern X, a primitive based constraint q and a dataset $\mathcal{D}$
Output: Interval condensed representation of constrained patterns having X as prefix
1: **if** $\neg PrefixFree(X)$ **then return** $\emptyset$ // *anti-monotone pruning*
2: **return** LocalScan$([X, \mathbf{cl}_{\preceq}(X)], q, \mathcal{D})$ // *local mining*
$\cup \bigcup \{\text{GlobalScan}(Xa, q, \mathcal{D}) | a \in \mathcal{I} \wedge a \geq \max_{\preceq} X\}$ // *recursive enumeration*

Algorithm 2. LocalScan

Input: An interval $[X, Y]$, a primitive based constraint q and a dataset $\mathcal{D}$
Output: Interval condensed representation of constrained patterns of $[X, Y]$
1: **if** $\lfloor q \rfloor \langle X, Y \rangle$ **then return** $\{[X, Y]\}$ // *positive interval pruning*
2: **if** $\neg \lceil q \rceil \langle X, Y \rangle$ **then return** $\emptyset$ // *negative interval pruning*
3: **if** $q(X)$ **then return** $[X, X] \cup \bigcup \{\text{LocalScan}([Xa, \mathbf{cl}_{\preceq}(Xa)], q, \mathcal{D}) | a \in Y \setminus X\}$
4: **return** $\bigcup \{\text{LocalScan}([Xa, \mathbf{cl}_{\preceq}(Xa)], q, \mathcal{D}) | a \in Y \setminus X\}$ // *recursive division*

Music-dfs scans the whole search space by running GlobalScan on each item of $\mathcal{I}$. GlobalScan recursively performs a depth-first search and stops whenever a pattern is not prefix-free (Line 1, GlobalScan). For each prefix-free pattern X, it computes its prefix-closed pattern and builds $[X, \mathbf{cl}_{\preceq}(X)]$ (Line 2, GlobalScan). Then, LocalScan tests this interval by using the operators $\lfloor . \rfloor$ and $\lceil . \rceil$ informally presented in Section 3.1. If the interval pruning can be performed, the interval is selected (positive pruning, Line 1 from LocalScan) or rejected (negative pruning, Line 2 from LocalScan). Otherwise, the interval is explored by recursively dividing it (Line 3 or 4 from LocalScan). The decomposition of the intervals is done so that each pattern is considered only once. The next theorem provides the correctness of Music-dfs:

Theorem 2 (Correctness). Music-dfs *mines soundly and completely all the patterns satisfying q by means of intervals.*

Proof. Property 2 ensures us that MUSIC-DFS enumerates all the interval condensed representation. Thereby, any pattern is considered (Theorem 1) individually or globally with the safe pruning stemmed from to the interval pruning (see Section 3.1). □

An additional anti-monotone constraint can be pushed in conjunction of prefix-freeness (Line 1, GLOBALSCAN). This constraint (e.g., minimal frequency constraint) optimizes the extraction by reducing more the search space. Such anti-monotone constraint is automatically deduced from the original constraint q in [21].

4 Mining Constrained Patterns from Transcriptomic Data

This section depicts the effectiveness of our approach on a transcriptomic case study. We experimentally show two results. First, the usefulness of the interval pruning strategy of MUSIC-DFS (the other prototypes fail for such large data, cf. Section 4.2). Second, BK enables to automatically focus on the most plausible candidate patterns (cf. Section 4.3). This underlines the need to mine constrained patterns by taking into account external data. If not mentioned otherwise, the experiments are run on the genomic data described in Section 4.1.

4.1 Gene Expression Data and Background Knowledge

In this experiment we deal with the SAGE (Serial Analysis of Gene Expression) [24] human expression data downloaded from the NCBI website (`www.ncbi.nlm.nih.gov`). The final binary dataset contains 11082 genes tested in 207 biological situations, each gene can be either over-expressed in the given situation or not. The biological details regarding gene selection, mapping and binarization can be seen in [10].

BK available in literature databases, biological ontologies and other sources is used to help to focus automatically on the most plausible candidate patterns. We have experimented with the gene ontology (GO) and free-text data. First, the available gene databases were automatically searched and the records for each gene were built (around two thirds of genes have non-empty records, there is no information available for the rest of them). Then, various similarity metrics among the gene records were proposed and calculated. More precisely, the gene records were converted into the vector space model [18]. A single gene corresponds to a single vector, whose components correspond to a frequency of a single term from the vocabulary. The similarity between genes was defined as the cosine of the angle between the corresponding *term-frequency inverse-document-frequency* (TFIDF) [18] vectors. TFIDF representation statistically measures how important a term is to a gene record. Moreover, the gene records were also simplified to get a condensed textual description. More details on text mining, gene ontologies and similarities are in [10].

4.2 Efficiency of Music-dfs

Dealing with large datasets Let us show the necessity of the depth-first search and usefulness of the interval pruning strategy of Music-dfs. All the experiments were conducted on a 2.2 GHz Xeon processor with 3GB RAM running Linux.

The first experiment highlights the importance of the depth-first search. We consider the constraint addressing patterns having an $area \geq 70$ (the minimal area constraint has been introduced in Section 2) and appearing at least 4 times in the dataset. Music-dfs only spends 7sec to extract 212 constrained patterns. In comparison, for the same binary dataset, the levelwise approach[1] presented in [20] fails after 963sec whenever the dataset contains more than 3500 genes. Indeed, the candidate patterns necessary to build the output do not fit in memory.

Comparison with prototypes coming from the FIMI repository (`fimi.cs.helsinki.fi`) shows that efficient implementations like kDCI [13], LCM (ver. 2) [23], COFI [25] or Borgelt's Apriori [3] fail with this binary dataset to mine frequent patterns occuring at least 4 times. Borgelt's Eclat [3] and Afopt [11] which are depth-first approaches, are able to mine with this frequency constraint. But they require a post-processing step for other constraints than the frequency (e.g., area, similarity-based constraints).

The power of Music-dfs can also be illustrated on any large benchmark dataset (i.e., containing many transactions). Let us consider the `mushroom` dataset taken from FIMI repository . Figure 3 presents the running times for the Music-dfs, Music, Apriori and Eclat algorithms with the constraints $freq(X) \times length(X) \geq \alpha$ (on the left) and $sum(X.val)/length(X) \geq \alpha$ (on the right). The latter is applied on item values (noted val) randomly generated within the range $[0, 100]$. An additional minimal frequency constraint $freq(X) \geq 100$ is used in order to make running of Apriori and Eclat feasible.

As Apriori and Eclat do not push the minimal area/average constraints into the mining, they require a post-processing step to select the right patterns

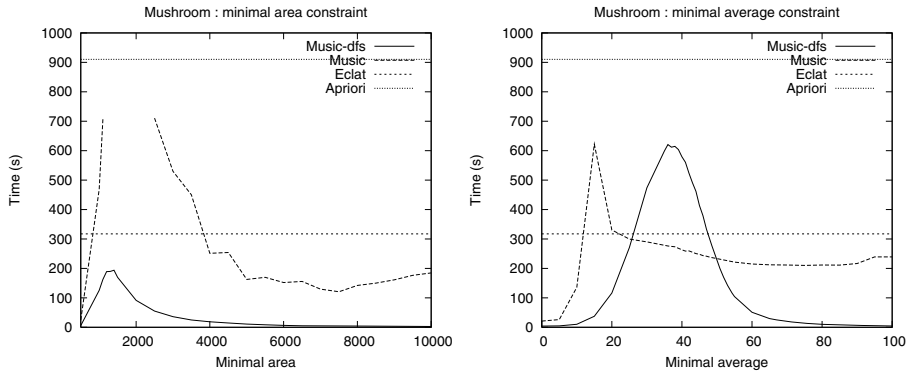


Fig. 3. Runtime performances with minimal area/average constraint on `mushroom`

[1] We do not use external data because this version does not deal with external data.

with respect to these constraints. Thus their curves (cf. Figure 3) do not depend on minimal area/average threshold α and are flat. Let us note that we neglect the time of the post-processing step therefore the total time spent by these methods is supposed to be even higher than shown. We observe that MUSIC-DFS clearly outperforms MUSIC and APRIORI. Moreover, MUSIC-DFS is often more efficient than ECLAT as it benefits from the constraint. The experimental study in [19] confirms that MUSIC-DFS is efficient with various constraints and various datasets.

Impact of interval pruning The next experiment shows the great role of the interval pruning strategy. For this purpose, we compare MUSIC-DFS with its modification that does not prune. The modification, denoted MUSIC-DFS-FILTER, mines all the patterns that satisfy the frequency threshold first, the other primitives are applied in the post-processing step. We use two typical constraints needed in the genomic domain and requiring the external data. These constraints and the time comparison between MUSIC-DFS and MUSIC-DFS-FILTER are given in Figure 4. The results show that post-processing is feasible until the frequency threshold generates reasonable pattern sets. For lower frequency thresholds, the number of patterns explodes and large intervals to be pruned appear. The interval pruning strategy decreases runtime and scales up much better than the comparative version without interval pruning and MUSIC-DFS becomes in the order of magnitude faster.

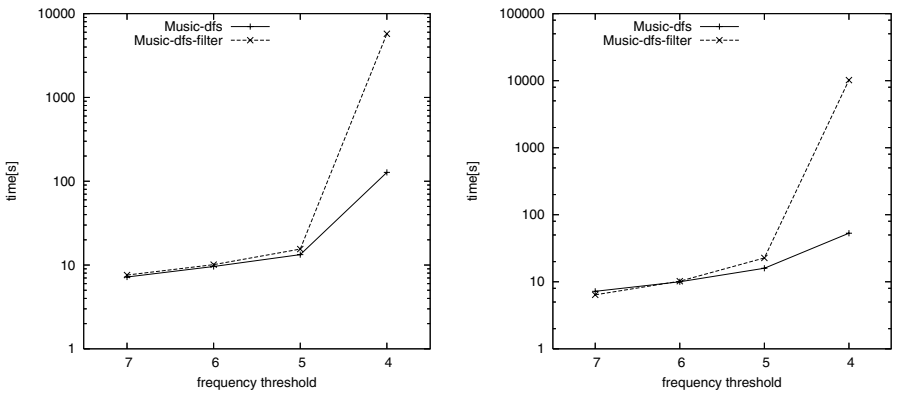


Fig. 4. Efficiency of interval pruning with decreasing frequency threshold. The left image deals with the constraint $freq(X) \geq thres \wedge lenght(X) \geq 4 \wedge sumsim(X)/svsim(X) \geq 0.9 \wedge svsim(X)/(svsim(X) + mvsim(X)) \geq 0.9$. The right image deals with the constraint $freq(X) \geq thres \wedge length(regexp(X, '*ribosom*', \texttt{GO\_terms})) = 0$.

4.3 Use of Background Knowledge to Mine Plausible Patterns

This transcriptomic case study demonstrates that constraints coming from the BK can reduce the number of patterns, they can express various kinds of interest and the patterns that tend to reappear are likely to be recognized as interesting

by an expert. One of the goals of any pattern is to generalize the individual gene synexpressions observed in the individual situations. Although it seems that biologists focus on individual biological situations, they follow very similar generalization goals. The most valuable knowledge is extracted from the patterns that concern genes with interesting common features (e.g., process, function, location, disease) whose synexpression is observed in a homogeneous biological context (i.e., in a number of analogous biological situations). An example of this context is the cluster of medulloblastoma SAGE libraries discovered in one of the constrained patterns (see the end of this section). It is obvious that to get such patterns and to pursue the goals mentioned above, a tool dealing with external data is needed.

Let us consider all the patterns having a satisfactory size which is translated by the constraint $area \geq 20^2$. We get nearly half a million different patterns that are joined into 37852 intervals. Although the intervals prove to provide a good condensation, the manual search through this set is obviously infeasible as the interpretation of patterns is not trivial and asks for frequent consultations with medical databases. The biologists prefer sets with tens of patterns/intervals only.

Increasing the threshold of the area constraint to get a reasonable number of patterns is rather counter-productive. The constraint $area \geq 75$ led to a small but uniform set of 56 patterns that was flooded by the ribosomal proteins which generally represent the most frequent genes in the dataset. Biologists rated these patterns as valid but uninteresting.

The most valuable patterns expected by biologists – denoted as meaningful or plausible patterns – have non-trivial size containing genes and situations whose characteristics can be generalized, connected, interpreted and thus transformed into knowledge. To get such patterns, constraints based on the external data have to be added to the minimal area constraint just like in the constraint q given in Section 2. It joins the minimal area constraint with background constraints coming from the NCBI textual resources (gene summaries and adjoined PubMed abstracts). There are 46671 patterns satisfying the minimal area constraint (the part (a) of the constraint q), but only 9 satisfy q. This shows the efficiency of reduction of patterns brought by the BK.

A cross-fertilization with other external data is obviously favourable. So, we use the constraint q' which is similar to q, except that the functional Gene Ontology is used instead of NCBI textual resources and a similarity constraint is added (part (e) of q').

$$
\begin{aligned}
q'(X) \equiv\ & area(X) \geq 24 && \text{(a)}\\
& \wedge\, length(regexp(X, '*ribosom*', \texttt{GO\_terms})) \leq 1 && \text{(b)}\\
& \wedge\, svsim(X, \texttt{GO})/(svsim(X, \texttt{GO}) + mvsim(X, \texttt{GO})) \geq 0.7 && \text{(c)}\\
& \wedge\, sumsim(X, \texttt{GO})/svsim(X, \texttt{GO}) \geq 0.025 && \text{(d)}\\
& \wedge\, insim(X, 0.5, 1, \texttt{GO})/svsim(X, \texttt{GO}) \geq 0.6 && \text{(e)}
\end{aligned}
$$

[2] This threshold has been settled by statistical analysis of random datasets having the same properties as the original SAGE data. First spurious patterns start to appear for this threshold area.

Only 2 patterns satisfy q'. A very interesting observation is that the pattern[3] that was identified by the expert as one of the "nuggets" provided by q is also selected by q'. This pattern can be verbally characterized as follows: it consists of 4 genes that are over-expressed in 6 biological situations, it contains at most one ribosomal gene, the genes share a lot of common terms in their descriptions as well as they functionally overlap, at least 3 of the genes are known (have a non-empty record) and all of the biological situations are medulloblastomas which are very aggressive brain tumors in children. The constraints q and q' demonstrate two different ways to reach a compact and meaningful output that can be easily human surveyed.

5 Conclusion

Knowledge discovery from a large binary dataset supported by heterogeneous BK is an important task. We have proposed a generic framework to mine patterns with a large set of constraints linking the information scattered in various knowledge sources. We have presented an efficient new algorithm MUSIC-DFS which soundly and completely mines such constrained patterns. Effectiveness comes from an interval pruning strategy based on prefix free patterns. To the best of our knowledge, there is no other constraint-based tool able to solve such constraint-based tasks.

The transcriptomic case study demonstrates that our approach can handle large datasets. It also shows practical utility of the flexible framework integrating heterogeneous knowledge sources. The language of primitives applied to a wide spectrum of transcriptomic data results in constraints formalizing a viable notion of interestingness.

Acknowledgements. The authors thank the CGMC Laboratory (CNRS UMR 5534, Lyon) for providing the gene expression database and many valuable comments. This work has been partially funded by the ACI "masse de données" (French Ministry of research), Bingo project (MD 46, 2004-07).

References

[1] Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proc. 20th Int. Conf. Very Large Data Bases, VLDB, pp. 432–444 (1994)
[2] Bonchi, F., Lucchese, C.: Pushing tougher constraints in frequent pattern mining. In: Ho et al. [7] pp. 114–124
[3] Borgelt, C.: Efficient implementations of Apriori and Eclat. In: Goethals, Zaki [6]
[4] Boulicaut, J.F., Bykowski, A., Rigotti, C.: Free-sets: a condensed representation of boolean data for the approximation of frequency queries. Data Mining and Knowledge Discovery journal 7(1), 5–22 (2003)

[3] The pattern consists of 4 genes KHDRBS1 NONO TOP2B FMR1 over-expressed in 6 biological situations BM_P019 BM_P494 BM_P608 BM_P301 BM_H275 BM_H876. BM stands for brain medulloblastoma.

[5] Bucila, C., Gehrke, J., Kifer, D., White, W.M.: Dualminer: A dual-pruning algorithm for itemsets with constraints. Data Min. Knowl. Discov. 7(3), 241–272 (2003)
[6] Goethals, B., Zaki, M.J. (eds.): FIMI '03, Frequent Itemset Mining Implementations, Proceedings of the ICDM 2003 Workshop on Frequent Itemset Mining Implementations, 19 December 2003, Melbourne, Florida, USA, CEUR Workshop Proceedings, vol. 90 (2003) CEUR-WS.org
[7] Ho, T.-B., Cheung, D., Liu, H. (eds.): Advances in Knowledge Discovery and Data Mining, PAKDD 2005. LNCS (LNAI), vol. 3518. Springer, Heidelberg (2005)
[8] Hébert, C., Crémilleux, B.: Mining frequent δ-free patterns in large databases. In: Hoffmann, A., Motoda, H., Scheffer, T. (eds.) DS 2005. LNCS (LNAI), vol. 3735, pp. 124–136. Springer, Heidelberg (2005)
[9] Jeudy, B., Rioult, F.: Database transposition for constrained (closed) pattern mining. In: Goethals, B., Siebes, A. (eds.) KDID 2004. LNCS, vol. 3377, pp. 89–107. Springer, Heidelberg (2005)
[10] Kléma, J., Soulet, A., Crémilleux, B., Blachon, S., Gandrillon, O.: Mining plausible patterns from genomic data. In: Lee, D., Nutter, B., Antani, S., Mitra, S., Archibald, J. (eds.) CBMS 2006, the 19th IEEE International Symposium on Computer-Based Medical Systems, Salt Lake City, Utah, pp. 183–188. IEEE Computer Society Press, Los Alamitos (2006)
[11] Liu, G., Lu, H., Yu, J.X., Wei, W., Xiao, X.: AFOPT: An efficient implementation of pattern growth approach. In: Goethals, Zaki [6]
[12] Mannila, H., Toivonen, H.: Levelwise search and borders of theories in knowledge discovery. Data Mining and Knowledge Discovery 1(3), 241–258 (1997)
[13] Orlando, S., Lucchese, C., Palmerini, P., Perego, R., Silvestri, F.: kDCI: a multi-strategy algorithm for mining frequent sets. In: Goethals, Zaki [6]
[14] Pan, F., Cong, G., Tung, A.K.H., Yang, Y., Zaki, M.J.: CARPENTER: finding closed patterns in long biological datasets. In: Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'03), Washington, DC, USA, pp. 637–642. ACM Press, New York (2003)
[15] Pasquier, N., Bastide, Y., Taouil, T., Lakhal, L.: Discovering frequent closed itemsets for association rules. In: Beeri, C., Bruneman, P. (eds.) ICDT 1999. LNCS, vol. 1540, pp. 398–416. Springer, Heidelberg (1998)
[16] Pei, J., Han, J., Lakshmanan, L.V.S.: Mining frequent item sets with convertible constraints. In: ICDE, pp. 433–442. IEEE Computer Society, Los Alamitos (2001)
[17] Rioult, F., Robardet, C., Blachon, S., Crémilleux, B., Gandrillon, O., Boulicaut, J.-F.: Mining concepts from large sage gene expression matrices. In: Boulicaut, J.-F., Dzeroski, S. (eds.) KDID, pp. 107–118. Rudjer Boskovic Institute, Zagreb, Croatia (2003)
[18] Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing Management 24(5), 513–523 (1988)
[19] Soulet, A.: Un cadre générique de découverte de motifs sous contraintes fondées sur des primitives. PhD thesis, Université de Caen Basse-Normandie, France, 2006 (to appear)
[20] Soulet, A., Crémilleux, B.: An efficient framework for mining flexible constraints. In: Ho„ et al. (eds.), [7] pp. 661–671 (2005)
[21] Soulet, A., Crémilleux, B.: Exploiting Virtual Patterns for Automatically Pruning the Search Space. In: Bonchi, F., Boulicaut, J.-F. (eds.) Knowledge Discovery in Inductive Databases. LNCS, vol. 3933, pp. 98–109. Springer, Heidelberg (2006)
[22] Stadler, B.M.R., Stadler, P.F.: Basic properties of filter convergence spaces (2002)

[23] Uno, T., Kiyomi, M., Arimura, H.: LCM ver. 2: Efficient mining algorithms for frequent/closed/maximal itemsets. In: Bayardo Jr., R.J., Goethals, B., Zaki, M.J. (eds.) FIMI. CEUR Workshop Proceedings, vol. 126 (2004), CEUR-WS.org
[24] Velculescu, V., Zhang, L., Vogelstein, B., Kinzler, K.: Serial analysis of gene expression. Science 270, 484–487 (1995)
[25] Zaïane, O.R., El-Hajj, M.: COFI-tree mining: A new approach to pattern growth with reduced candidacy generation. In: Goethals, Zaki [6]

Quantitative association rule mining in genomics using apriori knowledge

Filip Karel, Jiří Kléma

Department of cybernetics, Czech Technical University in Prague,
Technická 2, Praha 6, 166 27
karelf1@fel.cvut.cz, klema@labe.felk.cvut.cz

Abstract Regarding association rules, transcriptomic data represent a difficult mining context. First, the data are high-dimensional which asks for an algorithm scalable in the number of variables. Second, expression values are typically quantitative variables. This variable type further increases computational demands and may result in the output with a prohibitive number of redundant rules. Third, the data are often noisy which may also cause a large number of rules of little significance. In this paper we tackle the above-mentioned bottlenecks with an alternative approach to the quantitative association rule mining. The approach is based on simple arithmetic operations with variables and it outputs rules that do not syntactically differentiate from classical association rules. We also demonstrate the way in which apriori genomic knowledge can be used to prune the search space and reduce the amount of derived rules.

Keywords: association rules, quantitative attributes, apriori knowledge, SAGE

1 Introduction

At present, large quantities of gene expression data are generated. Data mining and automated knowledge extraction in this data belong to the major contemporary scientific challenges. For this task clustering is one of the most often used method [2] – the most similar genes are found so that the similarity among genes in one group (cluster) is maximized and similarity among particular groups (clusters) is minimized. Although very good results are gained by this method there are three main drawbacks [3]:

1. One gene has to be clustered in one and only one group, although it functions in numerous physiological pathways.
2. No relationship can be inferred between the different members of a group. That is, a gene and its target genes will be co-clustered, but the type of relationship cannot be rendered explicit by the algorithm.
3. Most clustering algorithms will make comparisons between the gene expression patterns in all the conditions examined. They will therefore miss a gene grouping that only arises in a subset of cells or conditions.

Association rule (AR) mining [1] can overcome these drawbacks. However, when dealing with datasets containing quantitative attributes it is often advisable to adapt the original AR mining algorithm. Mining of quantitative association rules (QARs) is considered as an interesting and important research problem. It was described in several papers such as [5], [6], [18], [19] which proposed various algorithmic solutions. Nevertheless, the proposed algorithms often do not take time consumption into the account.

QAR mining techniques aimed at gene-expression data were proposed for example in [4] or [15]. Half-spaces are used to generate QAR in [4], rules of the form 'if the weighted sum of some variables is greater than a threshold, then, with a high probability, a different weighted sum of variables is greater than second threshold'. An example of such rule can be '0.99 $gene_1$ - 0.11 $gene_2$ > 0.062 $\rightarrow$ 1.00 $gene_3$ > -0.032'. This approach naturally overcomes the discretization problem, on the other hand it is quite hard to understand the meaning of the rule.

In [15], the authors bring external biological knowledge to the AR mining. They mine rules which directly involve biological knowledge into the antecedent side of the rule. The given method can be applied to mine annotated gene expression datasets in order to extract associations like '$cell\_cycle \rightarrow [+]condition_1$, $[+]condition_2$, $[+]condition_3$, $[-]condition_6$', which means that, in the dataset, a significant number of the genes annotated as 'cell cycle' are over-expressed in condition 1, 2 and 3 and under-expressed in condition 6. This approach works with binary values of gene-expression only.

In this paper, QAR mining algorithm [12] is used and further developed. Despite it is very different from the classical AR algorithms, it outputs association rules in the classical form '$gene_i = <l\_value_{gi}..h\_value_{gi}> \wedge gene_j = <l\_value_{gj}..h\_value_{gj}> \wedge ... \rightarrow cancer = 0/1$. We can read this rule as 'when the value of $gene_i$ is between l_value_{gi} and h_value_{gi} and the value of $gene_j$ is between l_value_{gj} and h_value_{gj} and ... then with a high probability the cancer will (not) occur'. The task can be rephrased as search for the genes and their values that coincide with the appearance of cancer.

The algorithm is by no means limited to the particular right hand side (RHS) of rules. The target variable *cancer* is used here as it represents the most interesting outcome. The invariable RHS also simplifies the evaluation in Section 4. As follows from the structure of the rules, the presented algorithm deals with discretized quantitative attributes. A priori discretization influences resulting rules. One of the main interests of this paper is to compare the discretization into more bins (which prevents information loss) with binarization.

Background knowledge (BK) – the external apriori biological information – can be extracted using various publicly accessible web databases and tools [7], [8], [10]. Possibility of using this source of information to improve the generation of ARs is another aim of this paper. We show that appropriate implementation of BK can improve the quality of generated rules. The simplest utilization of BK is to give the rules their biological sense by straightforward annotation of the set of rules without their pruning. BK also helps to focus on specific rule subsets by early utilization of regular expressions. The most interesting use of BK is to

get the most plausible rules by application of gene similarity. Moreover, BK can significantly reduce the search space.

The paper is organized as follows: Section 2 presents the SAGE data, studies possible ways of its preprocessing and introduces apriori knowledge relevant to the given dataset. Section 3 gives an outline of QAR algorithm and discusses the ways it can employ apriori knowledge. Section 4 summarizes the reached results with the main stress on the effects of discretization and utilization of apriori knowledge. Finally we conclude in Section 5.

2 Character of SAGE data and preprocessing of raw data

The SAGE (Serial Analysis of Gene Expression) technique aims to measure the expression levels of genes in a cell population [20]. In this paper, the raw data matrix described in [11] was used. The expression dataset consists of 11082 tags (i.e., genes or attributes) whose expression was measured in 207 SAGE libraries (i.e. 207 biological situations or experiments). The tags represent the subset of human genome which is currently unambiguously identifiable by Identitag [3], the biological situations embody various tissues (brain, prostate, breast, kidney or heart) stricken by various possible diseases (mainly cancer, but also HIV and healthy tissues).

	$gene_1$	$gene_2$	...	$gene_n$	*cancer*
$situation_1$	0	15	...	0	0
$situation_2$	8	4	...	0	1
⋮	⋮	⋮	⋮	⋮	⋮
$situation_m$	3	0	...	39	1

Table 1. The structure of the raw SAGE data (n=11082, m=207), the gene values correspond to the expression of the particular gene in the particular biological situation, *cancer* stands for a binary class.

The structure of the raw SAGE expression dataset is in Table 1. As the main observed disorder is carcinoma, a target binary attribute *cancer* was introduced by the domain expert. The class value is 0 for all the healthy tissues and also the tissues suffering by other diseases than cancer (77 situations, 37.2%). It is equal to 1 for all the cancerous tissues (130 situations, 62.8%).

SAGE datasets are sparse – a great portion of gene-expression values equal to zero. The distribution of zeroes among genes is very uneven. Housekeeping genes are expressed (nearly) in all the tissues, however there is a reasonable amount of genes having zero values in almost all situations. Such genes are not suitable for further rule mining. Table 2 shows the numbers of frequently expressed genes. We can see that out of the total number of 11082 genes, only 97 have at least 95% non-zero values.

X	number of genes
5%	97
20%	305
50%	1038
80%	2703

Table 2. The number of genes having at the most X% of zero values

2.1 Discretization of expression values

In order to minimize the role of noise in SAGE data, the data are usually discretized first. As the discretization also brings the information loss, it is always disputable which type of discretization to apply. For a thorough discussion upon the impact of discretization see [16].

Binarization is now the most widely used method of discretization of gene expression data, where 0 means that the gene is under expressed and 1 means that the gene is over expressed. There are two disadvantages of data binarization: (1) it results in the biggest information loss, (2) it significantly influences (or rather forms) the output rules.

Table 3 describes the distinction among different types of binarization. 'Max -Y%' binarization means that the Y% of the highest value is the 0/1 threshold (provided the highest value of $gene_i$ is 100 and Y=90%, the threshold is 10, all the values above are encoded as 1). In 'median' binarization the border is the value of median. Logically, the most uniform distribution is obtained through the 'median' binarization. The most similar to 'median' is 'Max -80%' binarization using the gene sets with lower numbers of zeros values and 'Max -90%' using the gene sets with higher numbers of zero values.

	Max -90%	**Max -80%**	**Max -70%**	**Median**
X gene-set	**0/1 ratio**	**0/1 ratio**	**0/1 ratio**	**0/1 ratio**
5%	0.28 / 0.72	0.56 / 0.44	0.74 / 0.26	0.49 / 0.51
20%	0.32 / 0.68	0.59 / 0.41	0.77 / 0.23	0.49 / 0.51
50%	0.45 / 0.55	0.66 / 0.34	0.81 / 0.19	0.49 / 0.51
80%	0.60 / 0.40	0.74 / 0.26	0.84 / 0.16	0.61 / 0.39

Table 3. The results of binarization in terms of the 0/1 ratio. X defines the gene sets shown in Table 2.

Discretization into more bins enables more accurate rules. However, the classical equi-width and equi-depth approaches fail in this case. The former introduces intervals that are nearly empty, the latter keeps the same frequency across the intervals with unnatural bounds. The discretization based on 1-D clustering has to be employed. In short, the discretization steps repeated for each attribute are:

1. Initialize equi-distantly the *centers* of bins.
2. Assign every record value to the nearest center.
3. Recalculate every center position (average value of all records assigned to the center).
4. If the position of all centers did not move then end, else go to 2/.

The results of discretization into four and six bins are in Table 4. 4-bin discretization has approximately the same number of values assigned to the lowest bin as 'Max -80%'. Better resolution is obtained in higher values only. Using 6-bin discretization the resolution is better even in low values. But still low numbers of values are assigned to the higher bins. This is caused by the original distributions of gene expression values, where the majority of values is very close to zero.

	4-bin discretization	**6-bin discretization**
X gene-set	**1/2/3/4 ratio**	**1/2/3/4/5/6 ratio**
5%	0.63 / 0.24 / 0.08 / 0.05	0.45 / 0.27 / 0.13 / 0.06 / 0.06 / 0.03
20%	0.65 / 0.25 / 0.07 / 0.03	0.48 / 0.29 / 0.12 / 0.05 / 0.04 / 0.02
50%	0.69 / 0.23 / 0.06 / 0.02	0.52 / 0.27 / 0.10 / 0.04 / 0.05 / 0.01
80%	0.74 / 0.19 / 0.05 / 0.02	0.59 / 0.20 / 0.08 / 0.04 / 0.08 / 0.01

Table 4. The ratio of the number of values using the clustering discretization.

2.2 Background knowledge

Genomic websites such as NCBI [10] or EBI [9] offer a great amount of heterogeneous background knowledge available for various biological entities. In this paper we focused on Gene Ontology (GO) terms. To access the gene annotation data for every tag considered, RefSeq identifiers were translated into EntrezGene identifiers [8], the mapping approached 1 to 1 relationship. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the EntrezGene database [10] and sequentially parsed by Python scripts.

GO terms A list of related GO terms can be found for each gene (however for a certain portion of genes there are no GO terms available and the list is empty). This list characterizes the given gene and can be used to assume on its molecular function (MF) or the biological processes and the cellular components it participates in. The lists can be searched by regular expressions in order to focus on specific subsets of genes.

Similarity matrices GO terms can straightforwardly be used to compute similarity among genes. The rationale sustaining this method is that the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related. Two matrices – for BPs and MFs – created by authors in [11] are used. The structure of the gene similarity matrices is in

Table 5. The similarity values lie in the interval $< 0; 1 >$, where 1 stands for the genes with the identical description for the given category of terms. There are around 85% of missing similarity values (denoted n/a) for the genes with empty lists of related GO terms.

	$gene_1$	$gene_2$	$gene_3$	$gene_4$	...	$gene_n$
$gene_1$	0.15	0.75	n/a	...	n/a	
$gene_2$		n/a	0.12	...	0.93	
$gene_3$			0.64	...	n/a	
$gene_4$				...	n/a	
⋮					⋮	
$gene_n$						

Table 5. The structure of the gene similarity matrix.

In order to simplify the notion of similarity, both the above-described matrices are combined into one matrix as follows:

$$sim_{ij} = sim(BP)^2_{ij} + sim(MF)^2_{ij}$$

where $sim(BP)_{ij}$ is the similarity value for the genes i and j with respect to their biological process GO terms, $sim(MF)_{ij}$ is the similarity value for the same genes with respect to their molecular function GO terms.

3 QAR algorithm

An innovative QAR algorithm [12] is used for AR generation in this paper. The detailed algorithm description is out of the scope of this paper. The essential principles of the algorithm can be summarized as follows:

1. The input of the algorithm is a set of *atomic attributes*: $a_1, a_2, ...a_n$.
2. All the atomic attributes are discretized into D discretization bins and mapped to the consecutive row of integers beginning with one and ending with D (one represents the lowest value and D the highest value of an atomic attribute).
3. These preprocessed atomic attributes $pa_1, pa_2, ...pa_n$ are used to construct compound attributes – $x_i(pa_1, pa_2, ...pa_n) : N^n \rightarrow N$. *Compound attribute* is $x_i(pa_1, pa_2, ...pa_n) = \sum_{k=1}^{n} c_k a_k$, where c_k = {-1, 0, 1}, where i is number of compound attribute.
4. Each atomic (compound) attribute has a discrete distribution $P_i(t)$, two atomic (compound) attributes have a joint distribution $P_{ij}(t, s)$.
5. O is a set of all compact square or rectangle areas $o \subset <-\infty, \infty>$ x $<-\infty, \infty>$. For each pair $(x_i, x_j) \in P$ the algorithm searches for the best *areas of interest* o, where for each $(\alpha, \beta) \in o$

$$P_i(\alpha)P_j(\beta) - P_{ij}(\alpha, \beta) \geq \epsilon$$

6. From the areas of interest the best rules are extracted.

This algorithm takes an inspiration from earlier proposed algorithms [6], [14] or [19], but it comes with lower time consumption and pruning of redundant rules. On the other hand, the algorithm does not exhaustively enumerate all the relevant rules as it is not based on complete search through the state space. The algorithm works for binary attributes as well, although it loses its main advantages.

3.1 Injection of background knowledge into QAR algorithm

In order to increase noise robustness, focus and speed up the search, it is vital to have a mechanism to exploit background knowledge during AR generation. In the presented algorithm, BK can be taken into the account during the phase that combines atomic attributes into compound attributes.

The first option takes advantage of the lists of terms that describe the individual atomic attributes (genes in the SAGE data). The terms enable to focus on the rules that contain genes with specific characteristics. Provided x denotes a compound attribute, the variable $regexp$(x,'$*ribosom*$') delivers the number of genes that belong to x and whose at least one term matches the regular expression '$*ribosom*$'. The variable can be employed to get a limited set of rules that concern mainly (or only) ribosomal genes.

The second option exploits the gene similarity matrices [11]. This option focuses on plausible ARs, i.e., the rules that contain at least a certain portion of genes having common properties. The properties themselves do not have to be given by the user. An association rule can originate solely from the compound attributes with the value of gene similarity higher than a user defined threshold. Provided x denotes a compound attribute, the variable $svsim(x)$ gives the number of gene pairs belonging to x whose mutual similarity is known (distinct from n/a) and $mvsim(x)$ stands for its counterpart. $Sumsim(x)$ denotes the similarity sum over the set of genes belonging to x, $insim(x, min, max)$ stands for the number of gene pairs whose similarity lies between min and max.

Consequently, the variable $\frac{sumsim(x)}{svsim(x)}$ makes the average similarity of the compound attribute x, while the variable $\frac{insim(x,thres,1)}{svsim(x)}$ gives a proportion of the strong interactions (similarity higher than the threshold) within the compound attribute. The variable $\frac{svsim(x)}{svsim(x)+mvsim(x)}$ can avoid the compound attributes with prevailing genes of an unknown function. Relational and logical operators enable to create the final constraint, e.g., $V_1 \geq thres1$ and $V_2 \neq thres2$ where V_i stands for an arbitrary variable characterizing the compound attribute. Although we consider GO terms only, the framework is obviously general and the constraints can also be simultaneously derived from different external datasets.

The described technique obviously causes early pruning of the search space. Some of the compound attributes are rejected and the algorithm does not further search for the rules which do not satisfy the condition given by BK.

4 Experiments and results

This section presents the achieved experimental results. The influence of selected discretization methods is discussed. ARs in the classical form are generated. Conditions on the gene expression values are conjuncted on their LHS, the number of conditions is limited to three. The rules always have the attribute 'cancer' on their RHS. Confidence, support [1] and lift [17] measures are used to evaluate the quality of rules.

The file with maximum of 5% zero values was used. The input table for AR mining consists of 98 genes (attributes) and 207 situations (transactions). The number of attributes is low as the general scalability of the presented algorithm is not concerned here. It has already been proven in earlier works [12,13], along with its ability to reduce redundancy of the resulting set of rules. The main concern is to demonstrate applicability of BK to further improve understandability and scalability of QAR mining.

4.1 Rules without background knowledge

Table 7 shows the influence of discretization methods on the number of generated rules. This number is several times higher using a multi-bin discretization compared with binarization. There are also distinctions among particular binarization types, although not so significant. More rules are generated using binarizations with a more uniform distribution of zero and one values.

Similarity of rules generated by different discretization techniques was also examined, although it is hard to exactly compare different sets of rules. We considered two rules equal when all the antecedent genes, which occurred in the first rule also occurred in the second rule. For example, if genes with ID numbers 9, 13 and 82 occurr in the $rule_1$ and the same genes also occurr in the $rule_2$, then $rule_1 = rule_2$, no matter what values the genes take in the rules. The results are captured in Table 6, where the value on i-th column and j-th row is gained as

$$r_{ij} = \frac{number\_of\_rules_{i,j}}{number\_of\_rules_j},$$

where $number\_of\_rules_{i,j}$ is the number of rules generated both by the i-th type of discretization and by the j-th type of discretization and $number\_of\_rules_j$ is the total number of rules generated by the j-th type of discretization.

We can see that the ratios are quite low. It means that one can achieve a certain percentage of rules that agree in both types of discretization but quite a high number of rules is different. For example, when using 'Max -70%' and 'Max -80%' we gain approximately the same absolute number of rules from which only one fifth is equal. Also, '6-bin' discretization identifies only from 60% to 70% of rules identified using other types of discretization.

Experimentally it was found that these numbers depend on *min_supp* threshold. Lowering *min_supp* the ratios of 'identical' rules increase and higher numbers of similar rules are generated.

	Max -90%	Max -80%	Max -70%	Median	4-bin	6-bin
Max -90%	1	0.37	0.07	0.57	0.30	0.56
Max -80%	0.25	1	0.21	0.41	0.58	0.51
Max -70%	0.05	0.18	1	0.39	0.45	0.74
Median	0.26	0.29	0.23	1	0.48	0.61
4-bin	0.12	0.37	0.25	0.44	1	0.58
6-bin	0.15	0.20	0.25	0.35	0.36	1

Table 6. The number of the equal rules having 3 antecedent attributes generated by different discretization methods.

4.2 Using background knowledge (BK) for rules generation

Syntactically the same rules were generated with using BK, but a pruning condition was added. Using notation from Section 3.1, the applied conditions can be written as: 'generate rules with a compound attribute x only if $insim(x, 0.65, 2) \geq 1$'. It means that x is acceptable only if there is a pair of genes of x whose similarity is higher than the $min\_sim = 0.65$ threshold (at the same time it positively holds $svsim(x) \geq 2$). This condition early prunes the space of compound attributes and it is not only a rule filtering condition as for example min_conf condition.

	Max -90%	**Max -80%**	**Max -70%**	**Median**	**4-bin**	**6-bin**
3-ant (min_conf=0.9)	1 102	1 672	1 453	2 392	2 617	4 210
3-ant (min_conf=1.0)	88	33	15	90	126	65
3-ant (min_conf=0.8)	1 681	3 227	1 977	5 453	4 432	6 966
3-ant (min_conf=0.9)	150	152	117	317	247	360

Table 7. The number of rules created by different types of discretization without using background knowledge (top) and with background knowedge (bottom). Min_supp = 0.1, min_lift = 1.3, min_similarity = 0.65

	Binarization	**4-bin**	**6-bin**
without background knowledge	1.5×10^6	6.5×10^6	1.2×10^7
with background knowledge	1.7×10^5	7.1×10^5	1.3×10^6

Table 8. Number of verifications.

The number of rules (bottom part of table 7) is approximately 10 times lower than without using BK, the same holds for the number of verifications that the algorithm carries out. For $min\_conf = 0.8$ we obtain approximately the same number of rules as for $min\_conf = 0.9$ without BK. Time consumption remains

about ten times lower as the time-consumption of used algorithm does not depend on min_conf.

Further, the similarity of rules generated with and without BK is explored. In Table 9 we can observe the top 5 genes (top) and the top 5 pairs of genes (bottom) according to the number of their occurrences in rules.

without BK				with BK			
Max -80%	Median	4-bin	6-bin	Max -80%	Median	4-bin	6-bin
4	9	2	13	41	58	13	13
75	6	13	97	18	36	97	41
70	58	6	2	43	9	41	97
43	97	3	6	16	43	16	9
72	52	97	3	52	13	58	16
4-44	21-58	25-78	13-97	3-88	16-58	13-75	6-17
4-75	9-55	2-18	2-97	53-75	13-58	13-55	11-97
55-72	9-42	89-97	2-90	42-43	22-51	6-17	11-13
4-71	9-36	2-97	13-46	41-76	43-75	13-40	13-75
4-70	9-52	3-75	13-86	41-63	43-52	11-13	13-95

Table 9. Top 5 genes (top) and top 5 pairs (bottom) according to the number of occurrences in rules.

For '4-bin' and '6-bin' discretizations the top 5 gene lists are almost the same. Without BK, all of the 4-bin discretization top genes are also the top genes for 6-bin discretization. With BK this holds for 4 out of 5 genes. By contrast, for binarizations (both with and without BK) there is no overlap in the top gene lists. If we compare the gene lists of the identical discretizations with and without using BK, we observe that the multi-bin discretization and the 'median' binarization get the identical gene sets with and without BK.

For the top 5 pairs we have very similar observations as for the lists of top 5 genes. Generally, in the categories with and without BK the 4-bin and 6-bin discretizations are giving very similar results. 'Max -80%' and 'median' binarizations differentiate quite a lot. Between the two categories the most similar results are gained for 4-bin and 6-bin discretizations.

A more detailed comparison of particular gene occurrences in generated rules with and without BK is in Figure 1. Some of the genes have almost the same number of occurrences ($gene_{13}$), whereas other genes which have a very high number of occurrences using BK do not appear frequently in runs without application of BK ($gene_{41}$).

In general, the genes with prevalence of 'n/a' values in the similarity matrices are discriminated from the rules when using BK. However, a gene without annotation can still appear in a neighborhood of 'a strong functional cluster' of other genes. This occurrence then signifies its possible functional relationship with the given group of genes and it can initiate its early annotation. On the other hand,

the genes with extensive relationships to the other genes may increase their occurrence in the rules inferred with BK.

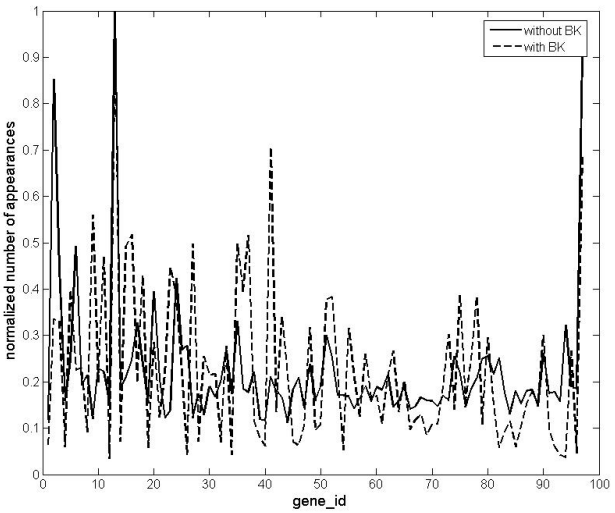


Figure 1. The frequency of particular genes in the generated rules with and without background knowledge for '6-bin' discretization.

5 Conclusions

In this paper, an alternative approach to QAR mining was verified on gene expression data. The paper discussed the influence of discretization methods on the generated rules. It was shown that the output set of rules is significantly influenced by the used discretization both wrt the number of generated rules and their composition. The presented QAR algorithm allowed us to use advantages of discretization into more bins and at the same time to generate rules without combinatoric explosion and without generation of redundant rules. In the light of our findings we think that more attention should be paid to the automatic discretization of gene expression values.

The paper also described and implemented the general framework for exploitation of BK during AR mining. It mainly helps to automatically focus on the most plausible candidate rules. At the same time, pruning conditions based on BK reduce time consumption significantly, while the number of plausible rules remains approximately the same. The conditions used in presented experiments were quite simple. Exploration of other possibilities of this framework and using more complex BK conditions is one of our major future challenges.

Acknowledgement. Filip Karel has been supported by the Ministry of Education, Youth and Sports of the Czech Republic as a part of the specific research

at the CTU in Prague - project nr. CTU0712613. Jiri Klema has been supported by the grant 1ET101210513 "Relational Machine Learning for Analysis of Biomedical Data" funded by the Czech Academy of Sciences.

References

1. R. Agrawal, T. Imelinsky, and A. Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD Conference on Management of Data*, pages 207–216, Washington, D.C., 1993.
2. Eisen M. B., Spellman P. T., Brown P. O., and Botstein D. Cluster analysis and display of genome-wide expression patterns. *Proceedings of National Academy of Science of the USA 95*, pages 14863–14868,, 1998.
3. Becquet C., Blachon S., Jeudy B., Boulicaut J-F, and Gandril O. Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data. *Genome Biology*, 3:531–537, 2002.
4. Georgii E., Richter L., Ruckert U., and Kramer S. Analyzing Microarray Data Using Quantitative Association Rules. *Bioinformatics*, pages ii123–ii129, 2005.
5. T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Mining optimized association rules for numeric attributes. In *In Proc. of ACM SIGMOD Conference on Management of Data*, Montreal, Canada, 1996.
6. S. Guillaume. Discovery of ordinal association rules. In *In Proceedings of the Sixth Pacific-Asia Conference PAKDD'02*, Taiwan, 2002.
7. http://crfb.univ mrs.fr/gotoolbox/. GOToolBox website.
8. http://discover.nci.nih.gov/matchminer/. Matchminer website.
9. http://www.ebi.ac.uk/. EBI website.
10. http://www.ncbi.nlm.nih.gov/. NCBI website.
11. Kléma J., Soulet A., Crémilleux B., Blachon S., and Gandrillon O. Mining plausible patterns from genomic data. In *19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)*, pages 183–190, 2006.
12. F. Karel. Quantitative and ordinal association rules mining (QAR mining). In *Knowledge-Based Intelligent Information and Engineering Systems*, volume 4251 of *LNAI*, pages 195–202. Springer, 2006.
13. F. Karel and J. Kléma. Ordinální asociační pravidla. In *Konference Znalosti 2005*, pages 226–233. VŠB-TUO, 2005.
14. R.J. Miller and Y. Yang. Association rules over interval data. In *In Proc. of ACM SIGMOD Conference on Management of Data*, Tuscon, AZ, 1997.
15. Carmona-Saez P., Chagoyen M., Rodriguez A., Trelles O., Carazo J.M., and Pascual-Montano A. Integrated analysis of gene expression by association rules discovery). *BMC Bioinformatics*, page 7:54, 2006.
16. Ruggero G. Pensa, Claire Leschi, Jérémy Besson, and Jean-François Boulicaut. Assessment of discretization techniques for relevant pattern discovery from gene expression data. In *BIOKDD*, pages 24–30, 2004.
17. G. Piatetsky-Shapiro. Discovery, analysis, and presentation of strong rules. In *in Knowledge Discovery in Databases*, Cambridge, 1991.
18. R. Rastogi and K. Shim. Mining optimized association rules with categorical and numeric attributes. *IEEE Trans. on KD Engineering*, 14(1), 2002.
19. R. Srikant and R. Agrawal. Mining quantitative association rules in large relational databases". In *In Proc. of ACM SIGMOD Montreal*, 1996.
20. Velculescu V., Zhang L., Vogelstein B., and Kinzler K. SAGE (Serial Analysis of Gene Expression). *Science*, page 270:484.7, 1995.

In Silico Biology 8 (2008) 157–175
IOS Press

Constraint-Based Knowledge Discovery from SAGE Data

Jiří Klémal[a,c], Sylvain Blachon[b], Arnaud Soulet[d], Bruno Crémilleux[a] and Olivier Gandrillon[b,*]

[a]*GREYC, CNRS UMR 6072, Université de Caen, Campus Côte de Nacre, F-14032 Caen Cédex, France*
[b]*Université de Lyon, Lyon, F-69003, France; Université Lyon 1, Lyon, F-69003, France; CNRS, UMR5534, Centre de génétique moléculaire et cellulaire, Villeurbanne, F-69622, France*
[c]*Department of Cybernetics, Czech Technical University in Prague, Technická 2, Prague 166 27, Czech Republic*
[d]*LI, Université de Tours, Place Jean Jaurès, F-41029 Blois, France*

Edited by H. Michael; received 2 October 2007; revised 30 January 2008; accepted 11 February 2008; published 5 April 2008

ABSTRACT: Current analyses of co-expressed genes are often based on global approaches such as clustering or bi-clustering. An alternative way is to employ local methods and search for patterns – sets of genes displaying specific expression properties in a set of situations. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate patterns which can hardly be further exploited. A timely application of background knowledge available in literature databases, biological ontologies and other sources can help to focus on the most plausible patterns only. The paper proposes, implements and tests a flexible constraint-based framework that enables the effective mining and representation of meaningful over-expression patterns representing intrinsic associations among genes and biological situations. The framework can be simultaneously applied to a wide spectrum of genomic data and we demonstrate that it allows to generate new biological hypotheses with clinical implications.

KEYWORDS: Functional genomics, SAGE, local pattern, background knowledge, gene ontology, biomedical literature, constraint

INTRODUCTION

The generation of very large gene expression databases by high-throughput technologies like microrarray [1] or SAGE [2] calls for similarly high-throughput exploration of possible functional links between genes and gene products. The link analysis is based upon similar expression properties, as well as possible relationships between co-expression patterns and sub-sets of biological situations.

Various techniques have been used for exploring such relationships, including global techniques like hierarchical clustering or K-means, or local pattern extraction such as association rule discovery (ARD) [3–6] or formal concepts [7].

Local patterns are groups of genes that harbor a specific expression property which can be over- or under-expression, either related to a single baseline (more/less expressed in situation A than in situation B) or related to the gene expression regime across multiple situations (more/less expressed in situation

*Corresponding author. Tel.: +33 47244 8190; Fax: +33 47243 2685; E-mail: gandrillon@cgmc.univ-lyon1.fr.

Electronic publication can be found in *In Silico Biol.* **8**, 0014 <http://www.bioinfo.de/isb/2008/08/0014/>, 5 April 2008.

A than across multiple other situations). They provide the biologist with a list of genes that, through the "guilt by association" hypothesis [8], are supposed to vary together due to a genuine biological principle, such as common function within the cell.

Extraction of local patterns is justified by the limitations of the global methods (see [3]) as well as by the need to explore gene-to-gene relationships that would be too subtle (i.e. occurring in too small a number of situations, or in very heterogeneous situations) for detection by global approaches.

One of the main drawbacks of every local pattern approach is the huge number of extracted patterns. This is especially true in noisy data, such as transcriptomic data. At least three research directions can be explored for solving this problem. The first one relies upon the use of fault-tolerant pattern extraction (see e.g. [9]) – a difficult task whose tractability is to date still uncertain. The second direction tries to regroup patterns through hierarchical clustering [10]. In this paper, we propose a third solution using external sources to introduce constraints that focus on the most meaningful patterns. Different types of sources can be used, including Gene Ontology and literature-based evidence extracted through text-mining.

A *constraint* is a function evaluating whether a pattern is interesting, and can be used to streamline the pattern search. Gene expression data represent a new challenge for constraint-based pattern mining since the overall complexity of exhaustive pattern search is exponential with the number of genes (i.e., items) which itself is typically large. A simple approach can be decomposed into two distinct steps. Firstly, to mine all potentially interesting patterns satisfying an anti-monotone constraint (e.g., the usual constraint of minimum frequency) because this class of constraints can be efficiently pushed (to eliminate irrelevant itemsets/sets of genes early and minimize the number of itemsets to be examined). Secondly, to filter the resulting set of patterns by the remaining constraints. However, this naïve filtering approach performed by an ordinary level-wise algorithm is intractable due to the huge number of patterns [11]. Existing scalable techniques [12,13] are limited to particular kinds of constraints (closed patterns, δ-free patterns). Integration of arbitrary background knowledge in the mining process in order to focus on the most plausible patterns requires more powerful data mining techniques.

Background knowledge is available in relational and literature databases, ontological descriptions and other sources. Its effective use in analysis and interpretation of expression data is a popular research topic nowadays. However, the main effort is aimed at clustering and consequent integration of biological knowledge into the statistical data analysis framework. Background knowledge is typically used to annotate the expression based clusters for statistically over-represented (or under-represented) terms or categories [14,15]. The same knowledge can also be employed to directly cluster genes [16] or to perform meta-clustering on pre-merged expression and external datasets [17]. Among the approaches distinct from clustering [18] converts gene annotations into relational logic features, while [19] uses text mining to filter the most promising disease gene candidates. Recently an ARD-based approach has been proposed in which the authors search for associations among several data sources based on co-occurrence [20]. The resulting rules express e.g. a relation between a metabolic pathway and gene over(under)-expression in a group of biological conditions.

In this paper we introduce and apply a more general depth-first search framework which is based on a rich declarative language of *primitive-based constraints* enabling effective internal *pruning* and a condensed output representation based on *intervals*. This framework is implemented within the constraint-based pattern mining tool MUSIC (Mining with a User-SpecifIed Constraint). The first version of the tool was described in [21], this paper extends it towards utilization of external sources and the depth-first search. We demonstrate that our procedure leads to a very effective reduction of the number of patterns, together with an "interpretation" of the patterns in the form of a list of words related to the function of the genes involved in the pattern. To the best of our knowledge, there is no other

Table 1
Examples of primitives and their values in the data mining context of Fig. 1

Primitives		Values
	Boolean matrix	
freq (X)	frequency of X	freq (ABC) = 2
length (X)	length of X	length (ABC) = 3
	Textual data	
regexp (X, RE)	items of X whose associated phrases match the regular expression *RE*	regexp (ABC,'*ion*') = (AC)
	Similarity matrix	
sumsim (X)	similarity sum over the set of item pairs of X	sumsim (ABC) = 0.13
svsim (X)	number of stated item pairs belonging to X	svsim (ABC) = 2
mvsim (X)	number of missing item pairs belonging to X	mvsim (ABC) = 1
insim (X, min, max)	number of item pairs whose similarity lies between min and max	insim (ABC, 0.07, 1) = 1

The table provides the meaning of primitives as well as their values in the context of Fig. 1.

constraint-based tool to efficiently discover patterns from large data under a broad set of constraints linking the information distributed in various knowledge sources. Using external constraints in the context of pattern mining as well as the integration of internal and external constraints are therefore the main contributions of this paper.

MATERIALS AND METHODS

Constraint-based pattern mining through several datasets

Usual data-mining tasks rarely deal with a single dataset. Often it is necessary to connect knowledge scattered in several heterogeneous sources. In constraint-based mining, the constraints should effectively link different datasets and knowledge types. In the domain of genomics, there is a natural need to derive constraints both from expression data and descriptions of the genes and/or biological situations under consideration. Such constraints require an analysis of various data types – transcriptome data and background knowledge may be stored in the boolean, numeric, symbolic or textual format. This section presents our framework (and the declarative language) enabling the user to set flexible and meaningful constraints.

Let us consider the genomic mining context given in Fig. 1. Firstly, the data involved include a boolean transcriptome dataset also called internal data where the items correspond to genes, the transactions represent biological situations and the binary values indicate gene over-expression. Secondly, external data – a similarity matrix and textual resources – are considered. They summarize background knowledge that contains various information on items (i.e., genes). This knowledge is transformed into a similarity matrix and a set of texts. Each field of the triangular matrix $s_{ij} \in$ [0,1] gives a similarity measure between the items i and j. The textual dataset provides a description of genes. Each row of this dataset contains a list of phrases characterizing the given gene. The mined patterns are composed of items of the internal data (the corresponding transactions are usually also added). The external data are used to further specify constraints in order to focus on meaningful patterns. In other words, the constraints may stem from all the datasets (see the example of q in Fig. 1, the experimental section provides examples of other constraints).

Let $\mathcal{I}$ be a set of items. A pattern is a non-empty subset of $\mathcal{I}$. $\mathcal{D}$ is a boolean matrix composed of patterns usually called transactions. The constraint-based mining task aims to discover all the patterns

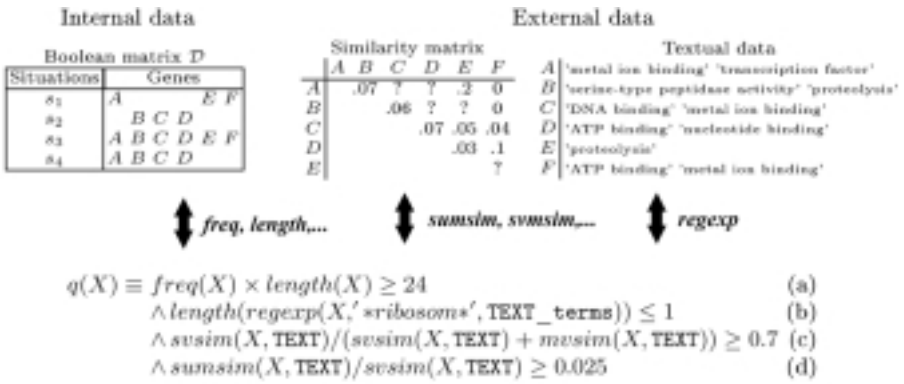


Fig. 1. An overview of constraint-based mining through several heterogeneous datasets. A toy example of the mining context along with a possible constraint. The figure shows various data types addressed by various sets of primitives. The constraint q addresses the large patterns (a) which are not composed of more than one ribosomal gene (b) and contain mainly annotated genes (c) with a minimal average similarity (d). The primitives are detailed in Table 1. The overall process can be viewed as a simultaneous query on data and on patterns. The combination of the primitive constraints can be therefore seen as an inductive query.

present in $\mathcal{D}$ and satisfying a constraint q. A pattern X is present in $\mathcal{D}$ whenever it is included in at least one transaction of $\mathcal{D}$. A distinctive point of our framework is its flexibility. Constraints are freely built of a large set of primitives representing a rich query language which allows to integrate various data/knowledge sources and to develop iteratively meaningful constraints.

Table 1 provides the meaning of the primitives involved in q and also in the other constraints used in this text. As primitives on external data are derived from different datasets, the dataset identification is another parameter of the primitive (for clarity not shown in Table 1). The first part (a) of q addresses the internal data and means that the biologist is interested in patterns having a satisfactory size (i.e., a minimal area). Indeed, *area* $(X) =$ *freq* $(X) \times$ *length* (X) is the product of the frequency of X and its length and means that the pattern must cover a minimum number of situations and contain a minimum number of genes. The other parts deal with the external data: (b) is used to discard ribosomal patterns (one gene exception per pattern is allowed), (c) to avoid patterns with prevailing items of an unknown function and (d) to ensure a minimal average similarity. Table 1 also indicates the values of these primitives in the context of Fig. 1. Our framework supports a large set of primitives, other examples of primitives with evident semantics are $\{\wedge, \vee, \neg, <, \leqslant, \subset, \subseteq, +, -, \times, /,$ *sum*, *max*, *min*, $\cup, \cap, \setminus\}$. The only theoretical property which is required on the primitives to belong to our framework is a property of monotonicity according to each variable of a primitive [21]. The constraints of this framework are called *primitive-based constraints*. Let us recall that the primitives and the constraints defined in [21] only address one boolean data set.

The framework is by no means restricted to the similarities and textual annotations discussed above. The requirement of monotonicity allows a wide range of data sources. In the genomic domain one can also implement constraints based directly on other resources such as interaction networks or lists of transcriptional regulators.

MUSIC tool and its efficiency

We use the tool MUSIC [21,22] which discovers soundly and completely all the patterns satisfying a given set of input constraints. The efficiency of MUSIC lies in its depth-first search strategy and a

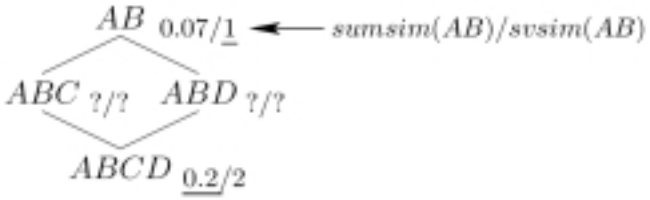


Fig. 2. Illustration of the interval pruning. The figure depicts an example of a pruning applied to the interval [*AB, ABDC*]. The pruning is exemplified with values of the primitives *sumsim* and *svsim*. The key idea is to exploit properties of the monotonicity of the primitives on the bounds of intervals. Whole intervals can be pruned at once.

safe pruning of the pattern space by pushing the constraints. The constraints are applied as early as possible. The pruning conditions are based on intervals representing several patterns. Whenever it is computed that all the patterns included in an interval simultaneously satisfy (or not) the constraint, the interval is positively (negatively) pruned without enumerating all its patterns [21]. The output of MUSIC enumerates the intervals satisfying the constraint. Such an interval condensed representation improves the output legibility and enables to easily compute the *selectivity* of the constraint. Selectivity is a proportion of patterns satisfying the constraint, and constitutes one of its important characteristics.

We start with the key idea of the safe pruning process based on intervals. The idea is to exploit properties of the monotonicity of the primitives on the bounds of intervals to prune them. This new kind of pruning is called *interval pruning*. Given two patterns $X \subseteq Y$, the interval [X,Y] corresponds to the set $\{Z \subseteq \mathcal{I} | X \subseteq Z \subseteq Y\}$. Figure 2 depicts an example with the interval [*AB, ABDC*] and the values of the primitives *sumsim* and *svsim*.

Assume the constraint *sumsim*(X)/*svsim*(X) $\geqslant$ 0.25. As the values associated to the similarities are positive, *sumsim*(X) is a function increasing with X. Thus *sumsim*(*ABCD*) is the highest *sumsim* value for the patterns in [*AB, ABCD*]. Similarly, all the patterns of this interval have a higher *svsim*(X) value than *svsim*(*AB*). Thereby, each pattern in [*AB, ABCD*] has its average similarity lower or equal than *sumsim*(*ABCD*)/*svsim*(*AB*) $=$ 0.2/1. As this fraction does not exceed 0.25, no pattern of [*AB, ABCD*] can satisfy the constraint and this interval can be pruned. We say that this pruning is *negative* because no pattern satisfies the constraint. In the same way, if the upper bound of the constraint on an interval [X, Y] increases the threshold, all the patterns in [X, Y] satisfy the constraint. [X, Y] is also pruned and this pruning is named *positive*. For instance, assuming that *sumsim*(*AB*)/*svsim*(*ABCD*) $\geqslant$ 0.02, then all the patterns in [*AB, ABCD*] satisfy the constraint.

In a more formal way, this approach is performed by two interval pruning operators $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ introduced in [21]. The main idea of these operators is to recursively decompose the constraint to take into account the monotone properties of the primitives and then to soundly prune intervals as depicted above. This process works straightforwardly with all the primitives tackling several kinds of datasets. This highlights the generic properties of our framework. Thereby, all the parts of the constraint q are pushed into the mining step.

Let us show the usefulness of the interval pruning strategy of MUSIC. The experiment was conducted on a 2.2 GHz Pentium IV processor with Linux operating system and 3GB of RAM memory. For this purpose, we compare MUSIC with its modification that does not prune. The modification, denoted MUSIC-filter, mines all the patterns that satisfy the frequency threshold first, the other primitives are applied in the post-processing step. We use two typical constraints needed in the genomic domain and requiring the external data. These constraints and the time comparison between MUSIC and MUSIC-filter are given in Fig. 3. The results show that post-processing is feasible until the frequency threshold

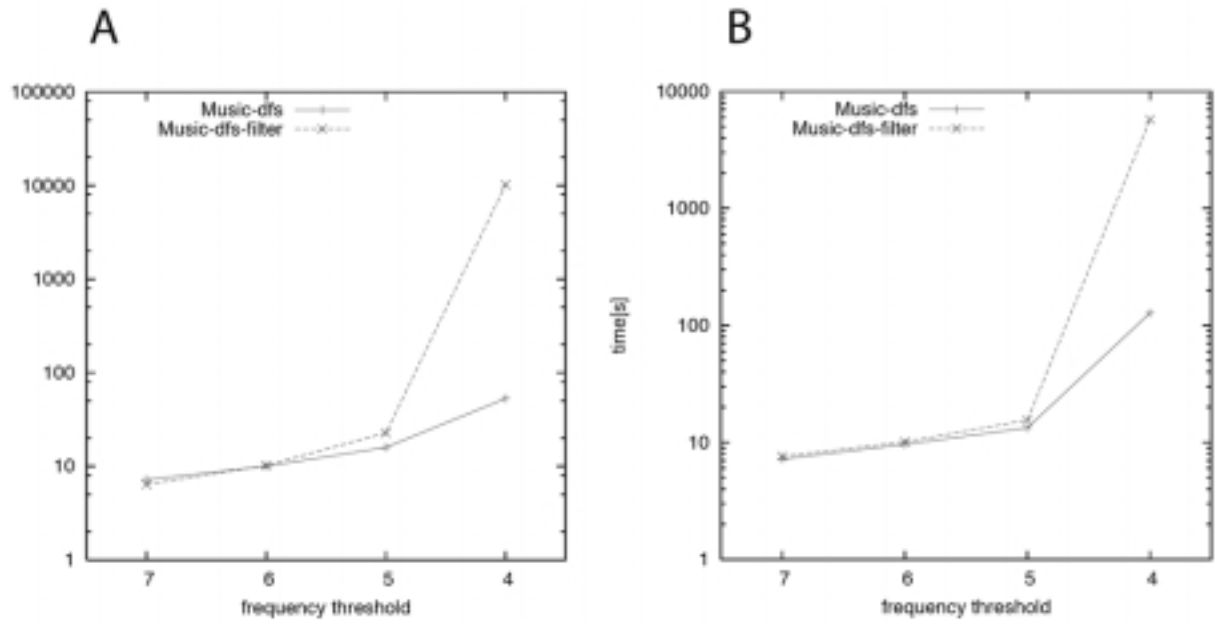


Fig. 3. Efficiency of the interval pruning. The efficiency of interval pruning with decreasing frequency primitive threshold is shown. The left image deals with the constraint $freq(X) \geqslant thres \wedge length(X) \geqslant 4 \wedge sumsim(X)/svsim(X) \geqslant 0.9 \wedge svsim(X)/(svsim(X) + mvsim(X)) \geqslant 0.9$. The right image deals with the constraint $freq(X) \geqslant thres \wedge length(regexp(X, \text{`}*ribosom*\text{'}, \text{GO\_terms})) = 0$.

generates reasonable pattern sets. For lower frequency thresholds, the number of patterns explodes and large intervals to be pruned appear. The interval pruning strategy decreases runtime and scales up much better than the comparative version without interval pruning and MUSIC becomes by orders of magnitude faster.

MUSIC prototype is available at http://www.info.univ-tours.fr/∼soulet/music-dfs/music-dfs.html.

MUSIC tool and its efficiency

The SAGE technique aims to measure the expression levels of genes in a cell population [2]. It is performed by sequencing tags (short sequences of 14 to 21 base pairs (bps) which are theoretically specific of each mRNA). A SAGE library is a list of transcripts expressed at one given time point in one given biological situation. Both the identity (assessed through a tag-to-gene complex process [23]) and the amount of each transcript is recorded. Analyzing such data is relevant since this SAGE data source has been largely under-exploited as of today, although it has the immense advantage over microarrays to produce datasets that can be directly compared between libraries without the need for external normalization. The human transcriptome can be seen as libraries that would be performed in each and every biologically relevant situations in the human body. This is clearly out of reach at the moment, and we deal in the present work with 207 very different situations ranging from embryonic stem cells to foreskin primary fibroblast cells. Biologists consider that useful knowledge about the transcriptome can be expressed as sets of genes and/or sets of biological situations that have some remarkable properties. Co-regulated genes, also known as synexpression groups, based on the guilt by association approach, are assumed to participate in a common function, or module, within the cell. The 207 SAGE libraries were downloaded from the NCBI web site as of October 2004 (http://www.ncbi.nlm.nih.gov/). To eliminate

putative sequencing errors, a pretreatment of the data described in [3] was applied, giving a set of 125,985 14 bp tags. Tags were identified thanks to Identitag [23], using RefSeq mRNA sequences. The unambiguous tags (displaying a 1 to 1 tag to RefSeq relationship) were selected, leaving a set of 11082 tags. A $207 \times 11{,}082$ gene expression matrix was built. There is also its sub-matrix which confines to the tags belonging to the minimal transcriptome [24]. It is based on 447 tags found and we refer to it as the minimum transcriptome (expression) matrix. To apply efficient local set pattern mining techniques on expression data, we must first identify and encode a specific gene expression properties (in principle, several properties per gene could be encoded, e.g., over-expression and under-expression). In this work, we decided to focus on over-expression. Thus if a gene is over-expressed in a situation then there will be a 1 value in the corresponding Boolean matrix cell, otherwise the value is 0. Both the matrices were binarized to encode the over-expression of each tag using the MidRange method described in [3]. For a thorough discussion upon the impact of discretization see [10,25].

Background knowledge

The section on constraint-based pattern mining introduces two principal kinds of external datasets, similarity matrices and textual files. The following three sections formalize the way in which they may be built. We use two principal external data sources, freetexts and gene ontologies (GOs), and preprocess them into the external datasets. In the area of freetexts we have been inspired mainly by [16,17]. Both of them deal with the term-frequency vector representation which is a simple however prevailing representation of texts. This representation allows for an annotation of a gene group as well as a straightforward definition of gene similarity. In the area of gene ontologies we stem from [15], the gene similarity results from the genes' positions in the molecular functional, biological process or cellular component ontology.

However, alternative sources can also be used, e.g. [26] suggests an approach to discover links between entities in biological databases. Information extracted from available databases is represented as a graph, where vertices correspond to entities and edges represent annotated relationships among vertices. A link is manifested as a path or a sub-graph connecting the corresponding vertices. Link goodness is based on edge reliability, relevance and rarity. Obviously, the graph itself or a corresponding similarity matrix based on the link goodness can serve as an external knowledge source.

Texts and their preprocessing

To access the gene annotation data for every tag considered, RefSeq identifiers were translated into EntrezGene identifiers (http://discover.nci.nih.gov/matchminer/). The mapping approached 1 to 1 relationship. There were only 11 unidentified RefSeqs, 24 RefSeqs mapped to more than 1 id and 203 ids appeared more than once. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the EntrezGene database (http://www.ncbi.nlm.nih.gov/) and sequentially parsed by the method stemming from [18]. The non-trivial textual records were obtained for 6302 ids which makes 58% of the total amount of 10,858 unique ids (3926 genes had a short summary, 5109 had one abstract attached at least).

The gene textual annotations were converted into the vector space model. A single gene corresponds to a single vector, whose components correspond to a frequency of a single term from the vocabulary. This representation is often referred to as *bag-of-words* [27]. The particular vocabulary consisted of all the *stemmed* terms (http://www.tartarus.org/∼martin/PorterStemmer/) that appear in 5 different gene records at least. The most frequent terms were manually checked and insufficiently precise terms (such as gene,

protein, human etc.) were removed. The resulting vocabulary consisted of 19,373 terms. The similarity between genes was defined as the cosine of the angle between the corresponding *term-frequency inverse-document-frequency* (TFIDF) [27] vectors. TFIDF representation statistically considers how important a term is to a gene record. A similarity matrix for all the tags was generated. The underlying idea is that a high value of two vectors' cosine (which means a low angle among two vectors and thus a similar occurrence of the terms) indicates a semantic connection between the corresponding gene records and consequently their presumable connection. Although this model is known to generate false positive relations for the sake of utilization of the same terms in a different context as well as false negative relations mainly because of synonyms, it is feasible and surprisingly often faithful.

Gene ontology

The genes can also be functionally related on the basis of their GO terms. The rationale sustaining this method is that the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related [15] defines a distance based on the Czekanowski-Dice formula, the methodology is implemented within the GOProxy tool of GOToolBox (http://crfb.univ-mrs.fr/GOToolBox/).

The original RefSeq tag identifiers were translated into UniProt ids[footnote]. Out of 11,082 tags there were 7670 known ids. As this set is too large to be processed by GOToolBox we confined the analysis to the minimum transcriptome dataset, 366 RefSeqs could be translated here. The resulting ids have been used by GoToolBox to generate two tag similarity matrices. For the biological process ontology there were 254 valid entries whereas 271 tags could be diagnosed within the molecular function ontology.

The GO terms themselves could be parsed from the records obtained in the previous subsection.

Description of libraries

There is a short textual annotation of about 10 terms attached to each SAGE library. Although these annotations represent very short documents, their vocabulary is quite compact. Consequently, they can be processed in the same way as the tag textual documentation. In this case, when considering all the terms that appear in 3 and more libraries the vocabulary consists of 83 terms. The situation similarity matrix was also generated.

This similarity matrix does not refer to items but transactions. The constraints are not inferred from it immediately but the matrix can be used in the latest phase of pattern annotation or filtration when the focus is on the most homogeneous transaction sets only.

RESULTS

General interaction among datasets

One of the basic questions rising prior to mining for the patterns is whether the datasets described above are mutually interconnected. Can we say that a group of tags that are functionally similar also tends to be co-expressed? Is there any relation between GO and textual definitions of similarity? Do similarly annotated situations tend to have similar expression profiles? Although the interconnection between the expression and external data is not a necessary condition to start the mining process, positive answers

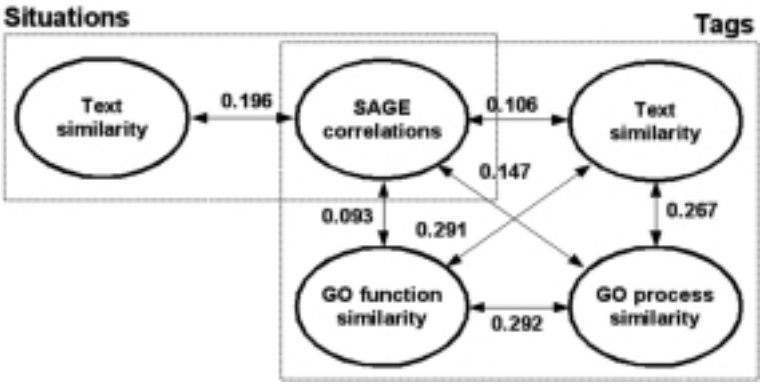


Fig. 4. Correlations among the datasets. The degree of correlation among the considered datasets. Similarity among gene profiles (or profiles of biological situations) is calculated within the individual datasets first. Then, the correlation between similarity matrices is determined. The higher the correlation between two datasets the more they agree in gene similarity. This experiment was performed on the minimum transcriptome matrix (207×447).

would support the overall logic of future experiments – the application of the similarity constraints should also lead to the compact expression data regions.

Correlation can serve as a general interconnection measure between expression and similarity data and also similarity datasets themselves. In order to get the matrices of the same dimension, the tag correlation matrix is derived from the expression data first. Then, its correlation with the tag similarity matrices is calculated. An analogical process is applied when dealing with the situations. Figure 4 shows that there is a statistically significant correlation among all the considered datasets. Nevertheless, the correlation values suggest a weak relationship only. When comparing the individual values, SAGE seems to be most strongly linked to the variance in situations. The interpretation may be such that SAGE deals with very different biological conditions – normal, cancerous or AIDS samples from different organs and individuals of different gender and age. They consequently vary in their expression profiles. The influence of tag similarity seems to be less striking. The similarity measure based on texts does not seem to be less valuable nor redundant with respect to the GO similarities.

Altogether this demonstrates the potential utility of using external sources for applying constraints, since all data sets are neither fully redundant, nor entirely disconnected.

How many patterns are statistically relevant?

One obvious source for noise in transcriptomic data lies within the experimental limitations of the techniques used. For example, SAGE is by essence a pooling strategy, and it has obvious limitations, especially for low to medium-sequenced libraries. Second, there is an intrinsic biological variation in the expression level of genes that has to be dealt with. Third, the binarization strategy cuts the expression values at a given threshold. Along with the use of formal concepts for generating patterns it can amplify the original experimental noise [28]. We therefore wanted to estimate the amount of patterns that were spurious, i.e., occurring randomly.

We generated 10 (pseudo)random datasets having the same properties as the original SAGE data: the same size ($11{,}082 \times 207$), the same density (the number of 1s is 53,511) and the same gene frequencies. The gene frequencies are very uneven, some of the genes are over-expressed in one situation only, others can be over-expressed in tens of situations. The roulette wheel technique [29] was used to keep the original gene frequencies. The generated datasets were searched for patterns of large areas. As the genes

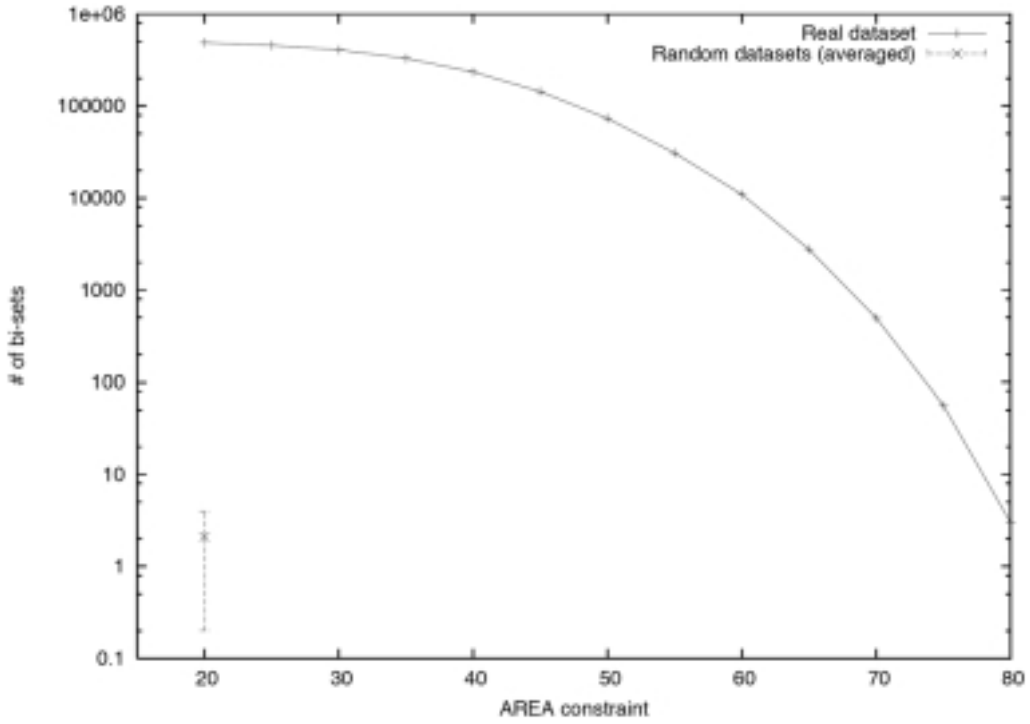


Fig. 5. Selectivity of the area constraint. The number of patterns larger than the given area. This experiment was performed on the complete 207 $\times$ 11,082 matrix.

are mutually independent, all of the patterns are necessarily spurious. Figure 5 shows their mean number as the function of the area and compares it with the number of patterns in the real dataset. The experiment proved that the random datasets contain no (spurious) patterns longer than 3 and more frequent than 5. The first spurious patterns (2.1 ± 0.9) tend to appear when the frequency threshold is decreased by one, i.e., the constraints are *length* $\geqslant 4$, *freq* $\geqslant 5$ and thus *area* $\geqslant 20$. These patterns contain exclusively the most frequent genes. In the real dataset we observe 490,267 patterns satisfying the same constraints. The experiment suggests that we may encounter at least about half a million non-random and thus large patterns.

The number of spurious patterns can also be theoretically estimated. Under assumption of gene independence and considering the prior frequency of genes, the probability that the pattern occurs at random is given by the multidimensional hypergeometric distribution:

$$p_s = l_{i=1}^{l} \frac{\binom{k_i}{f}\binom{m-k_i}{f-f}}{\binom{m}{f}} = l_{i=1}^{l} \frac{\binom{k_i}{f}}{\binom{m}{f}}$$

where l is the pattern length, f is the pattern frequency, m is the total number of situations and k_i is the frequency of i-th gene contained in the pattern. The probability p_s concerns specific biological context, i.e., it gives the chance that the pattern appears in a single set of situations. The total spurious occurrence

of the pattern can be estimated as follows:

$$n_s = \binom{m}{f} p_s = \binom{m}{f} l_{i=1}^{l} \frac{\binom{k_i}{f}}{\binom{m}{f}} = \frac{l_{i=1}^{l} \binom{k_i}{f}}{\binom{m}{f}^{I-1}}$$

The more real pattern occurrence exceeds n_s or the smaller its p_s, the more surprising and interesting pattern. The patterns of small area based on non-frequent genes can prove to be more interesting than their larger counterparts composed of the frequent genes. Consequently, the best internal constraint would be based on n_s or p_s, respectively. However, this constraint is difficult to calculate repeatedly during the pruning process. We have introduced it mainly to show that we deal with a large number of potentially meaningful patterns and they can be found even among patterns of a limited area.

The theoretical analysis confirmed that the final number of large patterns is even larger than mentioned in the experimental paragraph. Taken together these results clearly establish that the immense majority of the patterns that were generated could not by any means be attributed to noise, and have to be considered as potential source of biologically-relevant information. As the biologists prefer output sets with tens of patterns at most, one of the main tasks is to make this large number of potentially relevant patterns accessible to the expert in a friendly and interactive way.

Internal and external constraints to reach a meaningful limited pattern set

Since we have to deal with an explosion of putatively interesting relevant patterns, we tried to estimate the impact of applying various constraints during the extraction process. Traditional pattern mining deals with constraints that we refer to as internal. Their characteristic property is that they are inferred from the mined dataset. In our case, it is the binarized expression dataset. The main goal is usually to identify the itemsets (the sets of genes) that tend to co-occur frequently. The larger the itemsets, i.e., the more genes they contain, the better. Speaking of patterns, the most meaningful internal constraint regards their area, i.e., product of size/number of genes and frequency. It can be also understood as the number of ones that the pattern covers in the binarized expression dataset. Subjectively, the large patterns can be simply all the patterns that are larger than a certain threshold. However, we will define them as all the patterns that are large enough not to be spurious, i.e., occurring randomly. The goal is to find the optimal area threshold to distinguish between spurious and meaningful patterns.

Figure 5 shows how many patterns and intervals satisfy the increasing area constraint. In order to reduce the number of extracted patterns, the minimum pattern length was set to 4 and frequency to 5. Even using such a constraint, the number of patterns above a given area were still too numerous to be manually explored. For an example, there are 2090 intervals and 73,378 patterns having their area larger than 50. Let us note that the largest area patterns are very likely to be trivial, bringing no new knowledge, and it makes little sense to focus purely on them. At the same time, the selected binarization parameters generate rather sparse matrices. For other binarization types the explosion of patterns can be even faster.

There are two straightforward ways to treat the explosion of patterns. Firstly, one may try to focus on very large patterns only and increase the value of the area constraint. It is easy to show that this approach is rather counter-productive. The previous subsection on statistical pattern relevance clearly expresses that the more frequent genes are more likely to form very large patterns. In practice, the increase of the area threshold in order to get a reasonable number of patterns leads to a small but uniform set that is flooded by the ribosomal genes which represent the most frequent genes in our dataset. Biologists rated

these patterns as valid but since they were found earlier [3] they chose to discard them. Apparently, the area constraint helps to distinguish between spurious and real (random and non-random) patterns, but it does not hold that the larger the better. The pattern reduction by means of a stronger area restriction is unsound.

The second way relies upon a condensed representation of patterns. Comprehensibility increases as the human expert deals with fewer and more compact condensed sets of similar patterns. As the patterns tend to "overlap" greatly, let us try to test how far they can be condensed. Let us define the *maximal pattern* as such a pattern that no other pattern that satisfies constraints is its super-set. For example, having the set of patterns $s = \{b_1 = \{\{A, B, C\}, \{1, 2, 3\}\}, b_2 = \{\{A\}, \{2\}\}, b_3 = \{\{A, B\}, \{1, 2, 3\}\}, b_4 = \{\{A, C\}, \{1, 2, 4\}\}\}$, b_1 and b_4 are the maximal patterns while b_2 is a subset of b_1 and b_4, b_3 is a subset of b_1. Let us search for all the non-trivial patterns having *area* $\geqslant 24$. The search results in 46,671 different patterns which can be condensed into 2274 maximal patterns. It is fewer than the original number, but still too high to be manually inspected. Moreover, this maximal representation is incomplete and the original set of patterns cannot be restored from it. The interval condensed representation generated by MUSIC is complete, the number of intervals is usually higher than the number of maximal patterns (in this case we would have 9335 intervals).

Fundamentally different representation is a hierarchy of patterns [10]. The hierarchy is a result of clustering, whose partitions can speed up orientation among patterns, however, their number has to be decreased by external constraints again before the clustering is started. To sum up, the usual condensed representations of patterns are still too extensive to be surveyed by humans.

The previous paragraphs explain the motivation for using background knowledge to formalize constraints. It has been experimentally proven that the number of large patterns is so high that they cannot be effectively surveyed by a human expert. Simultaneous application of internal and external constraints, such as interestingness or expressiveness, may help to further reduce the patterns while keeping the interesting ones. The selectivity of selected external constraints is shown in Fig. 6. They capture the amount of similarity in given patterns through the measurement of the similarity of all tags pairs within that given pattern. *sumsim*(x)/*svsim*(x) expresses the average similarity, *insim* (x, *thres*, 1)/*svsim*(x) gives a proportion of the strong interactions (similarity higher than the threshold) within the set of tags, *svmsim*(x)/*svsim*(x + *mvsim*(x)) can avoid patterns with prevailing tags of an unknown function. The pruning starts with 46,671 patterns that are larger than 3 genes and more frequent than 5 libraries. The graphs depict that if both similarity (sumsim or insim) and existence (svsim) are thresholded, very compact sets of patterns can be reached. The next section gives a demonstration that these sets also gather biologically meaningful patterns.

Biological interpretation of patterns

The experimental setting started with all the large patterns that have a satisfactory average textual similarity among mostly known tags (see the measures *sim*1(x) $\geqslant 0.025$ and *sim*3(x) $\geqslant 0.7$ in Fig. 6. It was immediately apparent that most of the extracted patterns were harboring genes encoding ribosomal proteins, and proteins involved in the translation process. Such a trend has already been described, although in a different dataset [3], and we therefore decided to focus on some other biological functions. We further focused on patterns that did not harbor ribosomal proteins. This left us with a set of 19 patterns that were manually inspected. On the basis of their automatic explanation, we found the following pattern: B1 = {(*KHDRBS1*, *NONO*, *TOP2B*, *FMR1*) & (48, 52, 54, 56, 62, 65)}. There were 74 characteristic terms adjoined to genes, 8 terms characterized the situations. It is of biological interest for these reasons:

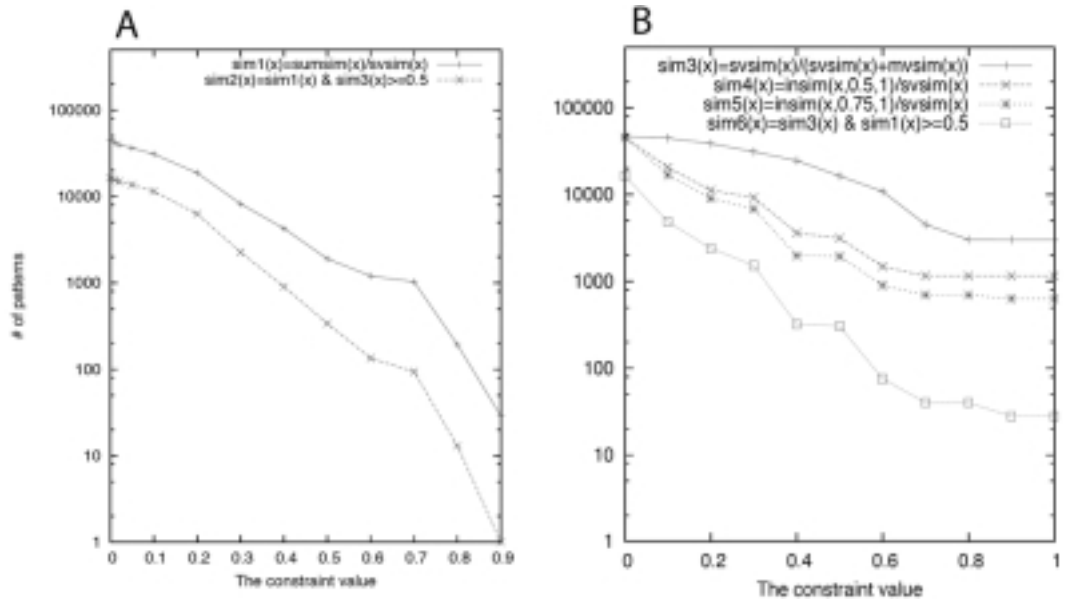


Fig. 6. Pattern pruning by the external constraints. Simultaneous application of internal and external constraints helps to arbitrarily reduce the number of patterns while attempting to conserve the potentially interesting ones. The figures show the decreasing number of patterns with increasing threshold of selected external constraints. The effect of six different constraints of various complexity is shown. This experiment was performed on the complete 207 $\times$ 11,082 matrix.

- Three out of the four genes (*KHDRBS1*, *NONO* and *FMR1*) have been shown to encode proteins that display an RNA-binding activity [30–32]. The term "RNA-bind" appears in the list of terms associated with this pattern. Of those genes, two (*KHDRBS1* and *NONO*) have been more specifically shown to be involved in RNA splicing.
- The fourth gene (*TOP2B*) encodes a topoisomerase [33]. It is interesting to note that the *NONO* gene product was shown to have a role in DNA unwinding [31], an activity where it is known to interact functionally with Topoisomerase 1 (a member of the family to which TOP2B belongs). Moreover an isoform of TOP2B, TOP2A, has also been found differentially expressed in medulloblastoma versus normal SAGE libraries [34]. The authors also note the existence of various anticancer drugs directed against TOP2A. These drugs might have an effect on the TOP2B isoform, enhancing the anticancer effect. A topoisomerase II inhibitor was also shown to display a significant antitumor activity in a medulloblastoma xenograft [35].
- A recent paper using a microarray has demonstrated the importance of RNA splicing processes for adult neurogenesis [36]. The *KHDRBS1* gene was found in this study among the genes important for adult neural stem cells.
- All of the situations in which these genes are over-expressed (48, 52, etc.) are medulloblastomas. These are very aggressive brain tumors in children. There is an increasing body of evidence that the most aggressive cells within a medulloblastoma behave as brain stem cells [37,38].

Altogether the biological hypothesis that can be made from this pattern is as follows: RNA binding in general and RNA splicing in particular, somehow connected with genomic DNA conformation via TOP2B, is as essential for medulloblastomas as it is for normal nervous system stem cells. Targeting this RNA binding activity, might prove beneficial for medulloblastoma treatment, just as topoisomerase II inhibition has proven to be.

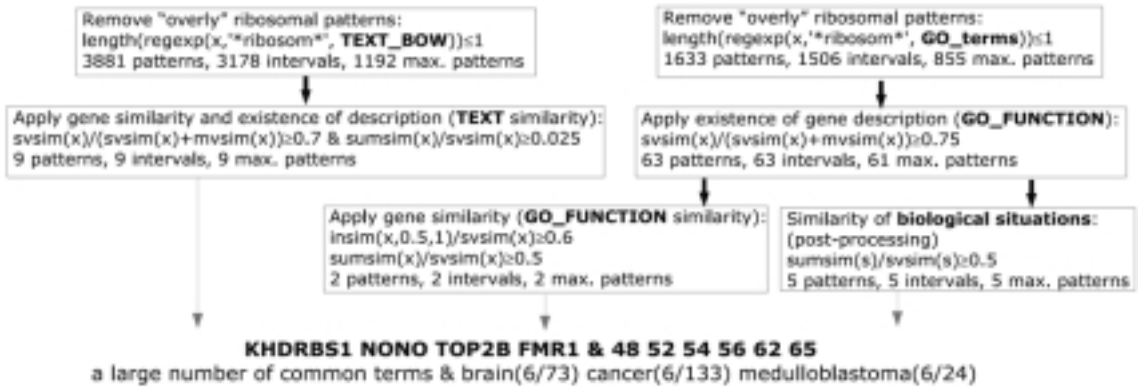


Fig. 7. Demonstration of selectivity and possible overlap among various constraints. The gradual reduction of patterns by background constraint is shown. The individual constraints are applied in conjunction. The figure demonstrates that background constraint can effectively reduce the number of patterns, it can define various domains of interest and the patterns that emerge are likely to be recognized as interesting by an expert. The example demonstrates three different ways to obtain a concise output that can be easily surveyed by a human because it consists of 9, 2 or 5 patterns only. An interesting observation is that the pattern that was identified by the expert as one of the "nuggets" (shown at the bottom of the image) can be obtained by several alternative ways. The first way uses NCBI textual resources (gene summaries and adjoined PubMed abstracts), the second way relies only on functional GO, while the third way utilizes similarities among biological situations too. Note that syntactically identical constraints aiming at textual and GO resources result in output of different quantity (3881 vs. 1633 patterns). Considering the datasets of different origin but the same format and purpose, the expert can decide whether to use them independently, unify or intersect them during pre-processing or via constraints. These experiments were performed on the complete 207x11082 matrix.

We then tried to assess the efficiency of using the GO-based external knowledge (annotations plus similarity), instead of the text-based one. We have constructed on principle a similar constraint to that mentioned at the beginning of this section. It is very interesting to note that the very same pattern that we previously analyzed (B1) was also found using this constraint. This clearly illustrated the level of redundancy that we previously described (see Fig. 7) and it demonstrates that some patterns are very robust.

We then focused on the following pattern: B2 = {(*EIF3S5*, *MRPL23*, *RPL18*, *EEF1G*) & (6, 30, 31, 116, 150, 171)}. This pattern is very homogeneous in term of the function of the genes since all of the genes participate to the translation machinery: *EIF3S5* encodes the eukaryotic translation initiation factor 3, *MRPL23* encodes the mitochondrial ribosomal protein L23, *RPL18* encodes the ribosomal protein L18 and *EEF1G* encodes the eukaryotic translation elongation factor 1 gamma. It is interesting to note that at the time we built our dataset, the gene *MRPL23* had no GO record attached. Therefore, it belongs to this pattern only by virtue of its expression pattern, although it encodes a mitochondrial ribosomal protein, and therefore also participates to the same function that the rest of the genes in this pattern. It is interesting to note that, although ribosomal genes were explicitly filtered out, one nevertheless obtained a pattern displaying such homogeneous, translation-related, functions. The nature of the situations harboring this set of simultaneously over-expressed genes is very heterogeneous, although some display stem cell characteristics (fibroblast cells immortalized by telomerase over-expression, CD34+ haematopoietic stem cells), and some do not (lung normal cell line). It is therefore difficult to understand why those situations have in common an over-expression of part of their translation machinery. One should nevertheless note that a preferential expression of translation-associated genes has just been described in murine haematopoietic stem cells [39]. In any case, this illustrates the power of local patterns to highlight gene expression patterns appearing though very different conditions, and that would not be captured by global tools like hierarchical clustering.

The example given in Fig. 7 gives another evidence that background constraints can effectively reduce the number of patterns, they can express various kinds of interest and the patterns that tend to reappear are likely to be recognized as interesting by an expert.

Gene function prediction

The proposed framework clearly serves knowledge discovery and the patterns correspond to descriptive models. Contrary to predictive models such as support vector machines they do not directly classify biological samples nor explicitly assign functions to genes. In this section we demonstrate an intelligible application of patterns for gene function prediction. Its motivation is twofold. Firstly, the descriptive models are hard to evaluate objectively. One can think of the manual evaluation of patterns done in the previous subsection as data fishing. The predictive experiment provides means to objectively assess the pattern sets en bloc. Secondly, the experiment implicitly outlines one of the ways the patterns can be interpreted by the biologists. On the other hand, the experiment does not outline the way to routinely and automatically predict gene functions. It is well known (see e.g. [40]) that similar gene expression profiles do not immediately imply similar tissue functions.

Let us assume the hypothesis that there is a relationship between the functional similarity of genes and their co-occurrence in patterns. Let us suppose that we have an expression dataset that mixes genes with known and unknown functions (annotations). Under our hypothesis, patterns can be applied to predict an unknown gene function in the following manner. Having a gene g with an unknown function, all the plausible patterns containing g are mined. The function of g is likely to relate to the function of the annotated genes that appear in the same patterns as g.

Let us experimentally verify our hypothesis. Obviously, gene co-occurrence in patterns does not imply gene functional similarity logically/immediately, the implication under consideration is probabilistic. That is why the predictive experiment tests all the genes that are frequent in the given expression dataset (and likely to appear in a sufficient number of patterns) and their annotation is known (the annotation is not used during pattern mining, only to evaluate the predictions). The hypothesis holds when the tested genes show a significantly higher functional similarity within their patterns than with other genes. The experiment pseudocode is as follows:

1. $\mathbb{E}$: $B \times G \rightarrow \{0, 1\}$ stands for a binary expression matrix, B is a set of m biological situations, G is a set of n genes, $\mathbb{S}$: $G \times G \rightarrow <0, 1> \cup \{NA\}$ is a gene similarity matrix (derived e.g. from the gene function ontology, *NA* stands for the undefined/missing similarity value).
2. Find a subset of frequent and annotated genes $F \subseteq G$ such that $F = \{f \in G |$ *freq*$(f) \geqslant$ *thres* $\wedge \exists i \neq f$: $S_{fi} \neq NA\}$, where $freq(f) = \sum_{b \in B} e_{bf}$. Frequent genes are likelier to appear in patterns, annotations are needed to make assumptions on the similarity among genes.
3. Select a minimum pattern frequency *pfreq* $\leqslant$ *thres*.
4. For each $f \in F$ calculate the weighted mean similarity to the other genes in the expression matrix:

$$\mathrm{msim}_f = \frac{\sum_{g \in G, \mathrm{freq}(g) > \mathrm{thres}}^{S_{fg} \neq NA, g \neq f} (\mathrm{freq}(g) - p\mathrm{freq}) S_{fg}}{\sum_{g \in G, \mathrm{freq}(g) > \mathrm{thres}}^{S_{fg} \neq NA, g \neq f} (\mathrm{freq}(g) - p\mathrm{freq})}$$

5. Choose a minimum pattern area *parea* $\geqslant$ *pfreq*. In $\mathbb{E}$ search for the set of all the large patterns *LPS* $\subseteq 2^G$ such that *LPS* $= \{P \subseteq G |$ *freq*$(P) \geqslant$ *pfreq* $\wedge$ *area*$(P) \geqslant$ parea$\}$, where *freq*$(P) =$ *supp*(P,E), *area*$(P) =$ *freq*$(P) \times$ *length*(P).

Table 2
Relation between gene co-occurrence in patterns and their similarity (in terms of molecular function and biological process)

Annotation type	F	$+/0/-$	msim	psim	p-value (paired t-test)
Molecular function	290	137/35/118	0.27	0.31	1.8E-7
Biological process	274	135/33/106	0.32	0.36	2.4E-8

The table shows mean similarity among genes. It averages over all the genes that are frequent enough (the expression matrix) and annotated (the similarity matrix). The value of *msim* estimates the similarity regardless patterns (it postulates that patterns do not correlate with gene annotation at all). The value of *psim* gives an estimate of the real gene similarity withinside patterns. The first row considers the similarity in terms of the molecular function, the second row concerns the biological process. The similarity is derived of the respective GO annotations. F is a number of the frequent and annotated genes, $+/0/-$ give numbers of genes out of F whose *msim* $>$ *psim*/*msim* $=$ *psim*/*msim* $>$ *psim*.

6. For each $f \in F$ find a subset LPS_f of large patterns *LPS* that contains f: $LPS_f = \{P \in LPS \mid f \in P\}$. Enumerate gene occurrence in LPS_f, every single occurrence of a gene is counted. GF_f is a set of gene occurrences in LPS_f such that: $GF_f = \{(g, g_{freq}) | g \in P \in LPS_f, g \neq f, g_{freq} = |\{P \in LPS_f | g \in P\}|\}$
7. For each $f \in F$ calculate the weighted mean similarity to the genes co-occurring in the large patterns:

$$\mathrm{psim}_f = \frac{\sum_{g \in \{(g, g_{\mathrm{freq}}) \in GF_f\}} g_{\mathrm{freq}} S_{fg}}{\sum_{g \in \{(g, g_{\mathrm{freq}}) \in GF_f\}} g_{\mathrm{freq}}} \quad (1)$$

8. Do a paired test between *msim* and *psim* vectors. The null hypothesis is that genes (the frequent and annotated) show no difference in their similarity to all the other genes and the genes that co-occur in their patterns. The alternative hypothesis states that the genes that co-occur in patterns tend to be more similar than randomly taken genes.

Table 2 summarizes the results for *thres* $= 15$, *area* $= 15$, *pfreq* $= 5$. It clearly shows that the intra-pattern functional gene similarity is significantly higher than the similarity among randomly sampled genes. The conclusion of this experiment is that the patterns actually generalize to the "unseen" cases, i.e., the patterns enable to draw attention to the function of yet unknown genes.

DISCUSSION

The goal of our work was to enhance the applicability of local pattern discovery for specific end users, such as biologists. For this we first verified that the immense majority of local pattern generated from human SAGE dataset were not attributable to random noise. This therefore clearly reinforces the need of automatic tools for navigating among the huge amount of potentially biologically relevant local associations among genes and situations.

We then verified that the external sources like Medline and Gene Ontology were at the same time sufficiently correlated and not too redundant so that their use would provide an add on value for selecting among the whole lists of patterns. We then applied a general filtering strategy based upon a new constraint-based mining algorithm, called MUSIC. Applying this algorithm on SAGE data could effectively lead to a very significant reduction in the amount of patterns the end user has to deal with. Furthermore, the "labeling" through lists of words rendered the selection of patterns for future exploration more easy.

Since the biological interpretation of a given pattern still has to be done manually, and is very time consuming, it is critical that such patterns are presented to the end-user in a way where he/she can choose rapidly which pattern is worth further investigation.

We applied this general strategy to a gene expression dataset displaying the expression of 11,082 genes in 207 different situations. We explored the patterns generated and found that some patterns are sufficiently robust to be generated through different types of constraints, either based upon GO-terms or upon text-based evidence. Compared to a recently published related work [20], our approach adheres to local patterns satisfying user-defined background properties specified by constraints. The fact that such constraints may be derived from current literature rather than through the use of an ontology makes it a more versatile tool, allowing recent evidence, available only in the literature, to be used as constraints.

One pattern obtained by the use of different constraints was further explored in detail. It led to an interesting hypothesis regarding the role of RNA-binding activities in the generation and/or maintenance of medulloblastomas. Another pattern pointed toward a role for the over-expression of part of the translation machinery in heterogeneous situations. Altogether this work demonstrates the usefulness of applying external constraints, and reinforces the potential impact of automated tools for analyzing large matrices of gene expression.

The predictive experiment confirmed the hypothesis that there is a relationship between the functional similarity of genes (and their products) and their co-occurrence in patterns. As a consequence, patterns enable us to draw attention to the function (and presumably other properties) of yet unknown genes.

In summary, constraints provide a human understandable way to extract valuable knowledge from potentially large and heterogeneous data. Provided they are computationally efficient, they enable interactive knowledge discovery resulting in the user-optimal set of constraints and consequently the set of desired patterns. We demonstrate the feasibility and usefulness of such an approach.

ACKNOWLEDGEMENTS

This work has been supported by the ANR (French Research National Agency) project BINGO2 ANR-07-MDCO-014 which is a follow-up of the first BINGO project (2004–007). The work of Jiří Kléma was partly funded by the Czech Ministry of Education in terms of the research programme Transdisciplinary Research in the Area of Biomedical Engineering II, MSM 6840770012. Sylvain Blachon was a fellow from the Comité de Saône et Loire de la Ligue Contre le Cancer. The work in Olivier Gandrillon's laboratory is supported by the Ligue contre le Cancer (Comité Départemental du Rhône), the UCBL, the CNRS, the Région Rhône Alpes (Thématique prioritaire) and the Association pour la Recherche contre le Cancer (ARC). Travels were covered by Czech-French PHC Barrande project "Fusion de données hétérogenes pour la découverte de connaissances en génomique". We thank Céline Keime for her help in identifying tags and for helpful discussions, Edmund Derrington (CGMC UMR 5534) for his critical and thoughtful reading of the manuscript and all members of the BINGO2 project for stimulating discussions.

After finishing the work for this manuscript, the server of University College London providing this functionality (previous URL: http://www.gene.ucal.ac.uk/nomenclature/data/gdlw/index.html) was removed from the net.

REFERENCES

[1] Gershon, D. (2002). Microarray technology: an array of opportunities. Nature **416**, 885-891.
[2] Velculescu, V., Zhang, L., Vogelstein, B. and Kinzler, K. (1995). Serial Analysis of Gene Expression. Science **270**, 484-7.
[3] Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J. F. and Gandrillon, O. (2002). Strong Association Rule Mining for Large Gene Expression Data Analysis: A Case Study on Human SAGE Data. Genome Biology **3**, 16 pages.
[4] Creighton, C. and Hanash, S. (2003). Mining gene expression databases for association rules. Bioinformatics **19**, 79-86.
[5] Georgii, E., Richter, L., Rückert, U. and Kramer, S. (2005). Analyzing microarray data using quantitative association rules. Bioinformatics **21** (**Suppl 2.**), ii123-ii129.
[6] Li, J., Liu, H., Downing, J. R., Yeoh, A. E.-J. and Wong, L. (2003). Simple rules underlying gene expression profiles of more than six subtypes of acute lymphoblastic leukemia (ALL) patients. Bioinformatics **19**, 71-78.
[7] Rioult, F., Robardet, C., Blachon, S., Crémilleux, B., Gandrillon, O. and Boulicaut, J. F. (2003). Mining Concepts from Large SAGE Gene Expression Matrices. *In*: KDID, Boulicaut, J.F., Dzeroski, S. (eds.), Rudjer Boskovic Institute, Zagreb, Croatia, pp. 107-118.
[8] Wolfe, C. J., Kohane, I. S. and Butte, A. J. (2005). Systematic survey reveals general applicability of "guilt-by-association" within gene coexpression networks. BMC Bioinformatics **6**, 227.
[9] Besson, J., Robardet, C. and Boulicaut, J. F. (2006). Mining a new fault-tolerant pattern type as an alternative to formal concept discovery. *In*: Proceedings of 14th International Conference on Conceptual Structures (ICCS'06), Aalborg, Denmark: Springer-Verlag, pp. 144-157.
[10] Blachon, S., Pensa, R., Besson, J., Robardet C., Boulicaut, J. F. and Gandrillon, O. (2007). Clustering formal concepts to discover biologically relevant knowledge from gene expression data. In Silico Biol. **7**, 0033.
[11] Jeudy, B. and Rioult, F. (2005). Database Transposition for Constrained (Closed) Pattern Mining. *In*: Post-Proceedings of the Workshop on Knowledge Discovery in Inductive Databases, Volume 3377 of Lecture Notes in Computer Science, Goethals, B. and Siebes, A. (eds.), Springer-Verlag, pp. 89-107.
[12] Pan, F., Cong, G., Tung A. K. H., Yang, Y. and Zaki, M. J. (2003). CARPENTER: finding closed patterns in long biological datasets. *In*: Proceedings of 9th ACM SIGKDD KDD conf., Washington, DC, USA, ACM Press, pp. 637-642.
[13] Rioult, F., Boulicaut, J. F., Cremilleux, B. and Besson, J. (2003). Using transposition for pattern discovery from microarray data. *In*: Proceedings of 8th ACM SIGMOD DMKD Workshop, San Diego, CA, pp. 73-79.
[14] Glenisson, P., Coessens, B., Van Vooren, S., Mathys, J., Moreau, Y. and De Moor, B. (2004). TXTGate: profiling gene groups with text-based information. Genome Biol. **5**, R43.
[15] Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D. and Jacq, B. (2004). GOToolBox : functional investigation of gene datasets based on Gene Ontology. Genome Biol. **5**, R101.
[16] Chaussabel, D. and Sher, A. (2002). Mining microarray expression data by literature profiling. Genome Biol. **3**, RESEARCH0055.
[17] Glenisson, P., Mathys, J. and Moor, B. D. (2003). Meta-clustering of gene expression data and literature-based information. SIGKDD Explor. Newsl. **5**, 101-112.
[18] Zelezny, F., Tolar, J., Lavrac, N. and Stepankova, O. (2005). Relational Subgroup Discovery for Gene Expression Data Mining. *In*: EMBEC: 3rd IFMBE European Medical & Biological Engineering Conf. Společnost biomedicinského inženýrství a lékařské informatiky čLS JEP, vol. 11.
[19] Tiffin, N., Kelso, J. F., Powell, A.R., Pan, H., Bajic, V. B. and Hide, W. A. (2005). Integration of text- and data-mining using ontologies successfully selects disease gene candidates. Nucleic Acids Res. **33**, 1544-1552.
[20] Carmona-Saez, P., Chagoyen, M., Rodriguez, A., Trelles, O., Carazo, J. M. and Pascual-Montano, A. (2006). Integrated analysis of gene expression by association rules discovery. BMC Bioinformatics **7**, 54.
[21] Soulet, A. and Crémilleux, B. (2005). An Efficient Framework for Mining Flexible Constraints. *In*: PAKDD, Volume 3518 of Lecture Notes in Computer Science, Ho, T.B., Cheung, D. and Liu, H. (eds.), Springer, pp. 661-671.
[22] Soulet, A., Kléma, J. and Crémilleux, B. (2007) Efficient Mining Under Rich Constraints Derived from Various Datasets. In: Knowledge Discovery in Inductive Databases, Dzeroski, S., Struyf, J. (eds.), Springer Berlin / Heidelberg, Volume 4747 of Lecture Notes in Computer Science, chap. 14, pp. 223-239.
[23] Keime, C., Damiola, F., Mouchiroud, D., Duret, L. and Gandrillon, O. (2004). Identitag, a relational database for SAGE tag identification and interspecies comparison of SAGE libraries. BMC Bioinformatics **5**, 143.
[24] Velculescu, V., *et al.* (1999). Analysis of human transcriptomes. Nat. Genet. **23**, 387-388.
[25] Pensa, R.G., Leschi, C., Besson, J. and Boulicaut, J.F. (2004). Assessment of discretization techniques for relevant pattern discovery from gene expression data. *In*: BIOKDD, Zaki, M.J., Morishita, S. and Rigoutsos, I. (eds.), pp. 24-30.
[26] Sevon, P., Eronen, L., Hintsanen, P., Kulovesi, K. and Toivonen, H. (2006). Link discovery in graphs derived from biological databases. *In*: Proceedings of 3rd International Workshop on Data Integration in the Life Sciences (DILS'06), Leser, U., Naumann, F. and Eckman, B. (eds.), LNBI 4075, pp. 35-49.
[27] Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing Management **24**, 513-523.

[28] Robardet, C., Pensa, R. G., Besson, J. and Boulicaut, J. F. (2004). Using Classification and Visualization on Pattern Databases for Gene Expression Data Analysis. *In*: PaRMa, Volume 96 of CEUR Workshop Proceedings, Theodoridis, Y. and Vassiliadis, P. (eds.), pp. 107-118.
[29] Baker, J. E. (1987). Reducing Bias and Inefficiency in the Selection Algorithm. *In*: Proceedings of the 2nd International Conference on Genetic Algorithms and their Application, Grefenstette, J. J. (ed.), Hillsdale, New Jersey, USA: Lawrence Erlbaum Associates, pp. 14-21.
[30] Lukong, K. E. and Richard S. (2003). Sam68, the KH domain-containing superSTAR. Biochim. Biophys. Acta **1653**, 73-86.
[31] Shav-Tal, Y. and Zipori, D. (2002). PSF and p54^{nrb}/NonO-multi-functional nuclear proteins. FEBS Lett. **531**, 109-114.
[32] Zalfa, F. and Bagni, C. (2004). Molecular insights into mental retardation: multiple functions for the Fragile X mental retardation protein? Curr. Issues Mol. Biol. **6**, 73-88.
[33] Champoux, J. J. (2001). DNA topoisomerases: structure, function, and mechanism. Annu. Rev. Biochem. **70**, 369-413.
[34] Boon, K., Edwards, J. B., Siu, I.-M., Olschner, D., Eberhart, C. G., Marra, M. A., Strausberg, R. L. and Riggins, G. J. (2003). Comparison of medulloblastoma and normal neural transcriptomes identifies a restricted set of activated genes. Oncogene **23**, 7687-7694.
[35] Vassal, G., Merlin, J.-L., Terrier-Lacombe, M.-J., Grill, J., Parker, F., Sainte-Rose, C., Aubert, G., Morizet, J., Sévenet, N., Poullain, M.-G., Lucas, C. and Kalifa, C. (2003). In vivo antitumor activity of S16020, a topoisomerase II inhibitor, and doxorubicin against human brain tumor xenografts. Cancer Chemother. Pharmacol. **51**, 385-394.
[36] Lim, D. A., Suárez-Fariñas, M., Naef, F., Hacker, C. R., Menn, B., Takebayashi, H., Magnasco, M., Patil, N. and Alvarez-Buylla, A. (2006) In vivo transcriptional profile analysis reveals RNA splicing and chromatin remodeling as prominent processes for adult neurogenesis. Mol. Cell. Neurosci. **31**, 131-148.
[37] Al-Hajj, M. and Clarke, M. F. (2004). Self-renewal and solid tumor stem cells. Oncogene **23**, 7274-7282.
[38] Derrington, E. A., Dufay, N., Rudkin, B. B. and Belin, M.-F. (1998). Human primitive neuroectodermal tumour cells behave as multipotent neural precursors in response to FGF2. Annu. Rev. Biochem. **17**, 1663-1672.
[39] Hüttmann, A., Dührsen, U., Heydarian, K., Klein-Hitpass, L., Boes, T., Boyd, A. and Li, C.-L. (2006). Gene expression profiles in murine hematopoietic stem cells revisited: analysis of cDNA libraries reveals high levels of translational and metabolic activities. Stem Cells **24**, 1719-1727.
[40] Yanai, I., Korbel, J. O., Boue, S., McWeeney, S. K., Bork, P. and Lercher, M. J. (2006). Similar gene expression profiles do not imply similar tissue functions. Trends Genet. **22**, 132-138.

Chapter XII
Discovering Knowledge from Local Patterns in SAGE Data

Bruno Crémilleux
Université de Caen, France

Arnaud Soulet
Université François Rabelais de Tours, France

Jiři Kléma
Czech Technical University in Prague, Czech Republic

Céline Hébert
Université de Caen, France

Olivier Gandrillon
Université de Lyon, France

ABSTRACT

The discovery of biologically interpretable knowledge from gene expression data is a crucial issue. Current gene data analysis is often based on global approaches such as clustering. An alternative way is to utilize local pattern mining techniques for global modeling and knowledge discovery. Nevertheless, moving from local patterns to models and knowledge is still a challenge due to the overwhelming number of local patterns and their summarization remains an open issue. This chapter is an attempt to fulfill this need: thanks to recent progress in constraint-based paradigm, it proposes three data mining methods to deal with the use of local patterns by highlighting the most promising ones or summarizing them. Ideas at the core of these processes are removing redundancy, integrating background knowledge, and recursive mining. This approach is effective and useful in large and real-world data: from the case study of the SAGE gene expression data, we demonstrate that it allows generating new biological hypotheses with clinical applications.

INTRODUCTION

In many domains, such as gene expression data, the critical need is not to generate data, but to derive knowledge from huge and heterogeneous datasets produced at high throughput. It means that there is a great need for automated tools helping their analysis. There are various methods, including global techniques such as hierarchical clustering, K-means, or co-clustering (Madeira & Oliveira, 2004) and approaches based on local patterns (Blachon et al., 2007). In the context of genomic data, a local pattern is typically a set of genes displaying specific expression properties in a set of biological situations. A great interest of local patterns is to capture subtle relationships in the data which are not detected by global methods and leading to the discovery of precious nuggets of knowledge (Morik et al., 2005). But, the toughness of extraction of various local patterns is a substantial limitation of their use (Ng et al.,1998; Bayardo, 2005). As the search space of the local patterns exponentially grows according to the number of attributes (Mannila &Toivonen, 1997), this task is even more difficult in *large* datasets (i.e., datasets where objects having a large number of columns).This is typically the case in gene expression data: few biological situations (i.e., objects) are described by ten of thousands of gene expressions values (i.e., attributes) (Becquet et al. 2002). In such situations, naive methods or usual level-wise techniques are unfeasible (Pan et al.,2003; Rioult et al., 2003). Nevertheless, especially in the context of transactional data, the recent progress in constraint-based pattern mining (see for instance (Bonchi & Lucchese, 2006; De Raedt et al., 2002) enable to extract various kind of patterns even in large datasets (Soulet et al., 2007). But, this approach has still a limitation: it tends to produce an overwhelming number of local patterns. Pattern flooding follows data flooding: the output is often too large for an individual and global analysis performed by the end-user. This is especially true in noisy data,such as genomic data where the most significant patterns are lost among too many trivial, noisy and redundant information. Naive techniques such as tuning parameters of methods (e.g., increasing the frequency threshold) limit the output but only lead to produce trivial and useless information.

This paper tackles this challenge. Relying on recent progress in constraint-based paradigm, it presents three data mining methods to deal with the use of local patterns by highlighting the most promising ones or summarizing them. The practical usefulness of these methods are supported by the case study of the SAGE gene expression data (introduced in the next section). First, we provide a method to mine the set of the simplest characterization rules while having a controlled number of exceptions. Thanks to their property of minimal premise, this method limits the redundancy between rules. Second, we describe how to integrate in the mining process background knowledge available in literature databases and biological ontologies to focus on the most promising patterns only. Third, we propose a recursive pattern mining approach to summarize the contrasts of a dataset: only few patterns conveying a trade-off between significance and representativity are produced. All of these methods can be applied even on large data sets. The first method comes within the general framework of removing redundancy and providing lossless representations whereas the two others propose summarizations (all the information cannot be regenerated but the most meaningful features are produced). We think that these two general approaches are complementary. Finally, we sum up the main lessons coming from mining and using local patterns on SAGE data, both from the data mining and the biological points of view. It demonstrates the practical usefulness of these approaches enabling to infer new relevant biological hypotheses.

This paper abstracts our practice of local patterns discovery from SAGE data. We avoid technical details (references are given for in-depth information), but we emphasize the main principles and results and we provide a cross-fertilization of our "in silico" approaches for discovering knowledge in gene expression data from local patterns.

MOTIVATIONS AND CONTEXT

Motivations

There is a huge research effort to discover knowledge from genomics data and mining local patterns such as relevant synexpression groups or characterization rules is requested by biologists. It is a way to better understand the role and the links between genes. Elucidating the association between a set of co-regulated genes and the set of biological situations that gives rise to a transcription module is a major goal in functional genomics. Different techniques including microarray (DeRisi et al., 1997) and SAGE (Velculescu et al., 1995) enable to study the simultaneous expression of thousands of genes in various biological situations. The SAGE technique aims to measure the expression levels of genes in a cell population. Analyzing such data is relevant since this SAGE data source has been largely under-exploited as of today, although it has the immense advantage over micro-arrays to produce datasets that can be directly compared between libraries without the need for external normalization. In our work, we use publicly available human serial analysis of gene expression SAGE libraries. We built a 207x11082 data set made up of 207 biological situations described by 11,082 gene expressions (i.e., a set of genes identified without ambiguous tags which will be useful for the techniques integrating the background knowledge) and a 90x27679 data set gathering 90 biological situations for 27,679 gene expressions (i.e., all the available transcriptomic information from these libraries).

As said in introduction, local pattern discovery has become a rapidly growing field (Blachon et al., 2007) and a range of techniques is available for producing extensive collections of patterns. Because of the exhaustive nature of most such techniques, the so-called local patterns provide a fairly complete picture of the information embedded in the database. But, as these patterns are extracted on the basis of their individual merits, this results in large sets of local patterns, potentially highly redundant. Moreover, the collections of local patterns represent fragmented knowledge and their huge size prevents a manual investigation. A major challenge is their combination and summarization for global modeling and knowledge discovery. It is a key issue because a useful global model, such a classifier or a co-clustering, is often the expected result of a data mining process. As well as their exhaustive nature and their ability to catch subtle relationships, summarizations of local patterns can capture their joint effect and reveal a knowledge not conveying by the usual kinds of patterns. The next section provides a few attempts in this general direction.

Related Work

Several approaches have been proposed to reduce the number of local patterns irrespective of their subsequent use. Examples include condensed representations (Calders et al., 2005), compression of the dataset by exploiting the Minimum Description Length Principle (Siebes et al., 2006) or the constraint-based paradigm (Ng et al., 1998; De Raedt et al., 2002). Constraints provide a focus that allows to reduce the number of extracted patterns to those of a potential interest given by the user. Unfortunately, even if these approaches enable us to reduce the number of produced patterns, the output still remains too large for an individual and global analysis performed by the end-user. Recently, two approaches appeared in the literature, which explicitly have the goal of combining and selecting patterns on the basis of their usefulness in the context of the other selected patterns: these pattern set discovery methods are constraint-based pattern set mining (De Raedt & Zimmermann, 2007), and pattern teams (Knobbe

& Ho, 2006). Constraint-based pattern set mining is based on the notion of constraints defined on the level of pattern sets (rather than individual patterns). These constraints capture qualities of the set such as size or representativeness. In the pattern team approach, only a single subset of patterns is returned. Pattern sets are implicitly ranked on the basis of a quality measure, and the best-performing set (the pattern team) is reported. Even if these approaches explicitly compare the qualities of patterns between them, they are mainly based on the reduction of the redundancy.

On the other hand, we think that it should be a pity to consider the summarization of local patterns only from the point of view of the redundancy. Local patterns can be fruitfully gathered for global modeling and knowledge discovery. Interestingly, such global models or patterns can capture the joint effect of local patterns such as co-classification performs. This approach is a way of conceptual clustering and provides a limited collection of bi-clusters. These bi-clusters are linked for both objects (i.e., biological situations) and attributes (i.e., genes). Tackling genomic data, Pensa et al. (Pensa et al., 2005) show that the bi-clusters of the final bi-partition are not necessary elements of the initial set of the local patterns. The bi-partition may come from a reconstruction of the biological situations and genes defining the local patterns. Except for particular kinds of local patterns (e.g., closed patterns (Blachon et al., 2007)), due to their large number of attributes, there are few works on discovery knowledge from SAGE data (Kléma et al.).

Constraint-Based Pattern Mining

As said in introduction, methods presented in this paper stem from recent progress in constraint-based paradigm. A constraint is a way to express a potential interest given by the user. Due to the huge search space of candidate patterns, a challenge is to push constraints in the core of the mining process by automatically inferring powerful and safe pruning conditions in order to get patterns satisfying a constraint. At least in transactional domains, there are now generic approaches to discover *local patterns* under constraints (De Raedt et al., 2002; Soulet & Crémilleux, 2005) even in large datasets (Soulet et al., 2007). A survey of the primitive-based framework (Soulet & Crémilleux, 2005) is provided below. This framework is at the basis of our method integrating background knowledge. We give now basic definitions used among the paper.

Let I be a set of distinct literals called *items*, an itemset (or pattern) corresponds to a non-null subset of I. These patterns are gathered together in the language L_I: $L_I = 2^I \setminus \emptyset$. A transactional dataset is a multi-set of patterns (i.e., transactions) of L_I. Each *transaction* is a database entry. More generally, transactions are called *objects* and items *attributes*. For instance, Table 1 gives a transactional dataset D with 8 objects o_1,..., o_8 (e.g., biological situations) described by 6 items A,…, F (e.g., gene expressions). This is a toy example which will be used throughout this paper. A value 1 for a biological situation and a gene expression means that this gene is over-expressed in this situation. In the SAGE data, each situation belongs to a class value (cancer versus no cancer) according to the biological origin of the tissue of the situation. For that reason, we divide D in two datasets D_1 and D_2 and a situation is labeled by the item C_1 (i.e., it belongs to D_1) or C_2 (i.e., it belongs to D_2).

Local patterns are regularities that hold for a particular part of the data. Let X be a pattern. We recall that the support of X in D denoted by $supp(X, D)$ is the proportion of objects in D containing X (we omit D when this data set is used by default). For instance, $supp(AB) = 3/8$. The constraint-based pattern mining framework D aims at discovering all the patterns of L_I satisfying a given predicate q, named *constraint*, and occurring in D. A well-known example is the *frequency* constraint focusing on

Table 1. Example of a transactional dataset

	$\mathcal{D}$							
	Gene expressions							
Situations	A	B	C	D	E	F		
o_1			1				C_1	
o_2	1	1		1		1	C_1	$\mathcal{D}_1$
o_3	1			1	1		C_1	
o_4	1	1		1			C_1	
o_5	1			1			C_2	
o_6	1	1	1			1	C_2	$\mathcal{D}_2$
o_7		1	1	1			C_2	
o_8			1		1		C_2	

patterns having a support exceeding a given minimal threshold *minsupp* > 0: *supp*(*X*, D) ≥ *minsupp*. For instance, *AB* is a frequent pattern with *minsupp* = 0.2. We will also use an absolute definition of the support , the frequency of *X* denoted *freq*(*X*) (*freq*(*X*, D) = *supp*(*X*,D) x |D|). As previously, we omit D when this data set is used by default. For instance, *freq*(*AB*) = 3. The frequency of the rule $X \rightarrow Y$ is *freq*(*XY*) and its *confidence* is *supp*(*XY*)/*supp*(*X*).

There are a lot of various constraints to evaluate the relevance of local patterns (Ng et al., 1998; Soulet & Crémilleux, 2005). The constraint-based paradigm also includes interestingness measures (the frequency is an example) to select local patterns. In the following, we will use the area of a pattern *area*(*X*): it is the frequency of a pattern times its length (i.e., *area*(*X*) = *freq*(*X*) x *count*(X) where *count*(X) denotes the cardinality of *X*. The area can be seen as the translation in the constraint paradigm of a synexpression group. For instance, the pattern *AB* (or *ABD*) satisfies the constraint *area*(*X*) ≥ 6 (as previously, if no data set is specified, it means that D is used). Emerging patterns (EPs) are another example. They are at the core of the summaries presented in the following. An EP is a pattern whose support strongly varies between two parts of a dataset (i.e., two classes),enabling to characterize classes (Dong & Li, 1999). The growth rate of *X* is $gr_i(X) = supp(X, D_i)/supp(X, D \backslash D_i)$. More formally, if we consider the two cancer and no cancer classes, a frequent emerging pattern X satisfies the constraint $supp(X,D) \geq minsupp \wedge (gr_{cancer}(X) \geq mingr \vee gr_{no\ cancer}(X) \geq mingr)$.

MINING A SYNTHESIS OF CLASSIFICATION RULES

There is an intense need of classification and classes characterization techniques to perform data mining tasks required on real-world databases. For instance, the biological situations in SAGE data are divided into two classes (cancer and no cancer) and biologists would like to better understand the relationships between the genes and these classes. For that purpose, we use the characterization rules previously introduced in (Crémilleux & Boulicaut, 2002). Thanks to a property of minimal premises, these characterization rules provide a kind of synthesis of the whole set of classification rules (i.e., all

the rules concluding on a class value). This result stems from the property of specific patterns, the δ-free patterns which are made of attributes without frequency relations between them (Boulicaut et al., 2003). Experiments (Crémilleux & Boulicaut, 2002) show that the number of characterization rules is at least an order of magnitude lower than the number of classification rules. Unfortunately, the method given in (Crémilleux & Boulicaut, 2002) does not run on large datasets such as the SAGE data. For that reason we have proposed a new method (Hébert et al., 2005) based on the extension of patterns (the extension of a pattern X is the maximal set of the objects containing X), because the extension has few objects in large databases. We give now a formal definition of these characterization rules (X and Y are patterns and C_i is an item referring to a class value).

Definition 1 (characterization rules): *Let minfreq be a frequency threshold, δ be an integer, a rule* $X \rightarrow C_i$ *is a characterization rule if there is no rule* $Y \rightarrow C_i$ *with* $Y \subset X$ *and a confidence greater than or equal to 1-(δ/minfreq).*

Given a frequency threshold *minfreq*, this definition means that we consider only the minimum sets of attributes (i.e., the minimal premises) to end up C_i, the uncertainty being controlled by δ. For instance, in our running example (Table 1), with δ = 1 and *minfreq* = 2, $C \rightarrow C_2$ is a characterization rule (there is one exception), but $CD \rightarrow C_2$ is not a characterization rule (it is covered by the previous rule). We argue that this property of minimal premise is a fundamental issue for classification. Not only it prevents from over-fitting but also it makes the characterization of an example easier to explain. It provides a feedback on the application domain expertise that can be reused for further analysis.

The value of δ is fundamental to discover relevant rules. With δ = 0, every rule must have a confidence value of 1 (i.e., *exact* rule). In many practical applications, such as the SAGE data, there are generally very few exact rules due to the non-determinism of the phenomena. We have to relax the condition on δ to accept exceptions (the more δ raises, the more the confidence decreases).

We developed the FTCminer prototype which extracts the sound and complete collection of frequent characterization rules (Hébert et al., 2005). FTCminer follows the outline of a level-wise algorithm (Mannila & Toivonen, 1997). Its originality is the use of the extension of patterns and that there is no generation phase of all the candidates at a given level since the candidates are generating one at a time. Thanks to these techniques, we are able to mine characterization rules even in large data sets whereas it was impossible before (Becquet at al., 2002; Hébert et al., 2005). Main results on SAGE data are given in the section on experiments.

INTEGRATING INFORMATION SOURCES SYNTHESIZING BACKGROUND KNOWLEDGE

This section sketches our approach to integrate background knowledge (BK) in the mining process to focus on the most plausible patterns consistent with pieces of existing knowledge. For instance, biologists are interested in constraints both on synexpression groups and common characteristics of the descriptions of the genes and/or biological situations under consideration. BK is available in relational and literature databases, ontological trees and other sources. Nevertheless, mining in a heterogeneous environment allowing a large set of descriptions at various levels of detail is highly non-trivial. There are various ways to interconnect the heterogeneous data sources and express the mutual relations among

the entities they address. We tackle this issue with the constraint paradigm. We think it is a promising way for such a work, the constraints can effectively link different datasets and knowledge sources (Soulet et al., 2007).

Our approach is based on the primitive-based constraints (Soulet & Crémilleux, 2005). There are no formal properties required on the final constraints and they are freely built of a large set of primitives. The primitives have to satisfy solely a property of monotonicity according to their variables (when the others remain constant). We showed that the whole set of primitive-based constraints constitutes a super-class of monotone, anti-monotone, succinct and convertible constraints (Soulet & Crémilleux, 2008). Consequently, the proposed framework provides a flexible and rich constraint (query) language. For instance, the product of two primitives *count*(*X*) x *freq*(*X*) may address the patterns having a certain minimum length (i.e., containing a minimum number of genes) and frequency (i.e., covering a minimum number of situations). We referred to it as *area*(*X*) above.

Furthermore, this framework naturally enables to integrate primitives addressing external data. Let us consider the transcriptomic mining context given in Figure 1. The involved data include a transcriptome dataset also called internal data as in our running example. External data - a similarity matrix and textual resources - summarize BK that contains various information on genes. Each field of the triangular matrix $s_{ij} \in [0,1]$ gives a similarity measure between the genes *i* and *j*. The textual dataset provides a description of genes. Details on the processing of textual resources within this approach and primitives tackling external data are given in another chapter of this book (Kléma & Zelezny). The mined patterns are composed of genes of the internal data, the corresponding objects are usually also noted (and possibly analyzed). The external data are used to further specify constraints in order to focus on meaningful patterns. In other words, the constraints may stem from all the datasets. The user can iteratively develop complex constraints integrating various knowledge types.

A real example of a constraint *q*(*X*) is given in Figure 1. The first part (a) of *q* addresses the internal data and means that the biologist is interested in patterns satisfying a minimal area. The other parts deal with the external data: (b) is used to discard ribosomal patterns (one gene exception per pattern is

Figure 1. Example of a toy (transcriptomic) mining context and a constraint

Internal data

Boolean matrix $\mathcal{D}$

	Gene expressions					
Situations	*A*	*B*	*C*	*D*	*E*	*F*
o_1			1			
o_2	1	1		1		1
o_3	1			1	1	
o_4	1	1		1		

External data

Similarity matrix

	A	*B*	*C*	*D*	*E*	*F*
A		.07	?	?	.2	0
B			.06	?	?	0
C				.07	.05	.04
D					.03	.1
E						?

Textual data

A	'metal ion binding' 'transcription factor'
B	'serine-type peptidase activity' 'proteolysis'
C	'DNA binding' 'metal ion binding'
D	'ATP binding' 'nucleotide binding'
E	'proteolysis'
F	'ATP binding' 'metal ion binding'

freq, count,... ***sumsim, svmsim,...*** ***regexp***

$$
\begin{aligned}
q(X) \equiv\ & freq(X) \times count(X) \geq 24 && \text{(a)}\\
& \wedge length(regexp(X, '*ribosom*', \texttt{TEXT\_terms})) \leq 1 && \text{(b)}\\
& \wedge svsim(X, \texttt{TEXT})/(svsim(X, \texttt{TEXT}) + mvsim(X, \texttt{TEXT})) \geq 0.7 && \text{(c)}\\
& \wedge sumsim(X, \texttt{TEXT})/svsim(X, \texttt{TEXT}) \geq 0.025 && \text{(d)}
\end{aligned}
$$

allowed), (c) avoids patterns with prevailing items of an unknown function and (d) is to ensure a minimal average gene similarity. The usefulness of such a constraint is shown in the section on experiments.

We have proposed a general prototype Music-dfs which discovers soundly and completely all the patterns satisfying the specified set of constraints (Soulet et al., 2007). Its efficiency lies in its depth-first search strategy and a safe pruning of the pattern space by pushing the constraints. Extractions in large data sets such as the SAGE data are feasible. Section on experiments demonstrates that our procedure leads to a very effective reduction of the number of patterns, together with an "interpretation" of the patterns.

RECURSIVE PATTERN MINING

This section outlines the recursive pattern mining framework and the discovery of the recursive emerging patterns (Soulet, 2007). The key idea is to repeat the pattern mining process on output to reduce it until few and relevant patterns are obtained. The final recursive patterns bring forward information coming from each mining step.

As often in mining constraint-based local patterns, the so-called collections of frequent emerging patterns (EPs) are huge and this hinders their uses. Several works address methods to reduce these collections by focusing on the most expressive ones (Bailey et al., 2002) (which are only present in one class) or by mining a lossless condensed representation (Li et al., 2007; Soulet et al., 2004). Nevertheless, these approaches do not reduce enough the number of mined patterns. Moreover, setting thresholds (i.e., *minsupp* or *mingr*) is often too subtle. Both the quantity and the quality of desired patterns are unpredictable. For instance, a too high threshold may generate no answer, a small one may generate thousands of patterns. Increasing thresholds to diminish the number of output patterns may be counterproductive (see the example with the area constraint in the section on experiments). Mining recursive patterns aims at solving these pitfalls.

In this work, we deal with frequent emerging patterns. Recursive emerging patterns (REPs) are the EPs which frequently occur within the outputted EPs according to the classes. The assumption is that these EPs are significant because the recursive mining process enables to synthesize and give prominence to the most meaningful contrasts of a dataset. A recursive emerging pattern *k*-summary (a REP *k*-summary, see Definition 2) provides a short description of the dataset constituted at most *k* REPs summarizing the contrasts according to the classes. It is produced by the generic recursive pattern mining framework: for each step, the previous mined patterns constitute the new transactional dataset. A first step mines all the frequent emerging patterns, as usual in the constraint-based pattern mining framework. Then the outputted EPs are joined to form a new dataset $D^2 = D^2_{\text{cancer}} \cup D^2_{\text{no cancer}}$. The EPs concluding on the class cancer (or no cancer) constitute the new sub-dataset D^2_{cancer} (or $D^2_{\text{no cancer}}$) and the process is repeated. This recursive process is ended as soon as the result becomes stable. At the end, we get at most *k* patterns brought forward information coming from each mining step. They summarize the main contrasts repeated through the outputs. From an abstract point of view, REPs can be seen as generalizations of emerging patterns. Main features on the method, (e.g., the theoretical convergence of recursive mining, number of steps) are given in (Soulet, 2007) and are not developed here because they are not crucial in practice.

For example, Table 2 depicts the mining of REPs from D (cf. Table 1) with *minsupp*=0.1 and *mingr*=2. Obviously, the datasets D^2_1 and D^2_2 are exactly the EPs in $D = D^1$ with *minsupp*=0.1 and *mingr*=2. At the

Table 2. REPs mined from D with minsupp = 0.1 and mingr = 2

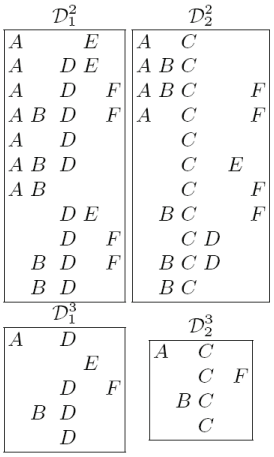


Table 3. REP 10-summary of D with mingr = 2

REPs of $\mathcal{D}_1$

REP	*supp*	gr_1
AD	0.5	3
E	0.25	1
DF	0.125	∞
BD	0.375	2
D	0.625	1.5

REPs of $\mathcal{D}_2$

REP	*supp*	gr_2
AC	0.125	∞
CF	0.125	∞
BC	0.25	∞
C	0.5	3

next mining step, the number of REPs (i.e., union of D^3_1 and D^3_2 : 9 patterns) is lower than the number of EPs (i.e., union of D^2_1 and D^2_2 : 22 patterns). In this example, EPs in D^4 are exactly the patterns of D^3 and then the collection of frequent REPs is stable: final REPs come from D^3 . We define below a REP *k*-summary which straightforwardly stems from REPs:

Definition 2 (REP *k*-summary): *A REP k-summary (according to mingr) is the whole collection of REPs obtained with minsupp=1/k and mingr.*

We argue that a REP *k*-summary is a compact collection of EPs having a good trade-off between significance and representativity. We proved (Soulet, 2007) that the size of a REP *k*-summary is bounded according to *minsupp*: to get at most *k* patterns in a REP *k*-summary, it is enough to fix *minsupp* = 1/*k*. For instance, the 10-summary in Table 3 contains 9 patterns (we have $9 \leq 10$ with *k*=10=1 /*minsupp* = 1/0.1). Moreover, we claim that it is easier for a user to fix a maximal value for the number of patterns than the support threshold.

Besides a REP *k*-summary covers a large part of the dataset D: most objects support at least one EP of the summary. This is due to REPs are frequent patterns in the dataset of each step. Thus, they are representative of the original dataset D, but also of all the emerging patterns from D. Table 3 recalls the REP 10-summary with *mingr*=2 from our running example. Supports (column *supp*) and growth rates (column gr_i) in the initial dataset D are added. As *minsupp*=1/10, this summary is exactly the REPs given in Table 2. Interestingly, we note that the growth rates of the REPs may be lower than *mingr* (e.g., $gr_1(D,\mathrm{D})$=1.5 whereas *mingr* = 2). This avoids the crisp effect of a threshold where a promising pattern is deleted only because its value for the measure is just under the threshold. The power of the recursive mining approach relies on the summarization: most of the REPs have a significative growth rate and all the objects (except o_1 and o_5) are covered by a REP concluding to their class values. Clearly, o_1 is closer to the objects of D_2 than objects of D_1, this explains why o_1 is not characterized by a REP. A similar reasoning can be done with o_5.

The tunable concision of REPs favours users' interpretation. Each REP can be individually interpreted as usual EPs, providing a qualitative and quantitative information. Appropriately, the small collection of REPs offers a global and complementary description of the whole dataset.

LESSONS FROM MINING SAGE DATA

The section outlines the main results achieved on SAGE data thanks to the previous data mining methods. Then, we synthesize the major lessons both from the data mining and the biological points of view.

A Sketch of Biological Results

To fully understand the results of experiments, we have to precise that each attribute of a SAGE data set is a *tag*. The identification of genes is closely related to the tags and biologists are able to associate genes and tags. In the case of the 207x11082 data set, each tag is unambiguously identified. This property is very useful to link together the information coming from several sources of BK.

Gene expressions are quantitative values and we must identify a specific gene expression property to get binary value and run the data mining methods depicted above. In principle, several properties per gene could be encoded, e.g. over-expression and under-expression. In our studies, we decided to focus on over-expression (over-expression has been introduced in the beginning of the paper). Several ways exist for identifying gene over-expression (Becquet et al., 2002). Results given in this paper are performed by using the mid-range method: the threshold is fixed w.r.t. the maximal value (*max*) observed for each tag. All the values which are greater than (100 - X%) of *max* are assigned to 1, 0 for the others (here, $X = 25$). For the 90x27679 data set, the values of tags vary from 0 to 26021. The percentage of tags which values are different from 0 is 19.86% and the arithmetic mean is around 4. As already said, the biological situations are divided into two classes (cancer and no cancer). 59 situations are labelled by cancer and 31 by no cancer (i.e., normal).

Characterization rules. We give the mean features on our work on mining characterization rules on SAGE data (more experiments and details are provided in (Hébert et al., 2005). In this paper, we only deal with the classes cancer and no cancer. More fruitful further biological investigations will require to use sub-groups of these classes, such sub-groups being defined according to biological criteria (e.g., a cancer type).

Table 4 presents a selection of rules with at least two tags in their body and a rather high confidence and frequency with *minfreq* and δ=1. Table 5 provides the description of tags (identification number, sequence and description) only for the tags which appear the most frequently in our results. Some tags are identified by several genes: their identifications are separated by ";".

Few tags (e.g., 4602, 8255, 11115, 22129) clearly arise in many rules concluding on cancer. They may have an influence on the development of this disease. It is interesting to note that the frequencies of these tags strongly varies from one class to another. For example, the tag 11115 appears 28.7 times more in rules characterizing cancer than no cancer. The tag 11115 is identified as GPX1. The expression of GPX1 has been found in various studies to be correlated with cancerous situations (Korotkina et al., 2002; Nasr et al., 2004). On the contrary, the tag 22129 appears 22 times more in rules concluding on no cancer than concluding on cancer. It might mean that this tag is related to normal development. We will come back on this tag below, with regard to the interestingness of biological results.

Table 4. Examples of potential relevant rules with minfreq = 4 and δ= 1

Premise	Conclusion	Exceptions	Frequency	Confidence
11115 19811	cancer	1	13	0.92
5961 11115	cancer	0	12	1
8279 23600	cancer	1	12	0.92
10960 11115	cancer	1	12	0.92
11115 20766	cancer	1	12	0.92
4602 7259 18882	cancer	1	10	0.9
4602 7259 24686	cancer	1	10	0.9
8255 11115 19811	cancer	1	10	0.9
4602 7259 20461	cancer	1	9	0.89
4602 7259 25202	cancer	1	9	0.89
4602 18882 24686	cancer	1	9	0.89
4287 4602 7818	cancer	1	8	0.88
4287 4602 19811	cancer	1	8	0.88
4602 7259 19734	cancer	1	8	0.88
4602 24686 25202	cancer	1	8	0.88
4602 25128 25202	cancer	1	8	0.88
7259 12667 16807	cancer	1	8	0.88
8255 11115 13642	cancer	0	8	1
8255 11115 26846	cancer	1	8	0.88
8255 19811 26846	cancer	1	8	0.88
22619 25202 26846 27358	cancer	1	5	0.8
16786 26715	no cancer	1	7	0.86
22129 25356	no cancer	1	7	0.86
22129 27414	no cancer	1	7	0.86
22647 25356	no cancer	1	7	0.86
1722 25202 26715	no cancer	1	6	0.83

Table 5. Characteristics of potential relevant tags

Number	Sequence	Description
4287	AGCTCTCCCT	RPL17 ribosomal protein L17
4602	AGGCTACGGA	Similar to ribosomal protein L13a, 60S ribosomal protein L13a, 23 kD highly basic protein
8255	CATCCAAAAC	HNRPH1 Heterogeneous nuclear ribonucleoprotein H1 (H)
11115	CTCTTCGAGA	GPX1 Glutathione peroxidase 1
19811	GTTGCTGCCC	NIFIE14 Seven transmembrane domain protein
22129	TCAGAGAATA	SLC25A22 Solute carrier family 25 (mitochondrial carrier: glutamate), member 22; IRS2 Insulin receptor substrate 2
25202	TGTGCTAAAT	RPL34 Ribosomal protein L34

Integrating BK. A highly valuable biological knowledge comes from the patterns that concern genes with interesting common features (e.g., process, function, location, disease) whose synexpression is observed in a homogeneous biological context (i.e., in a number of analogous biological situations). We give now an example of such a context with the set of medulloblastoma SAGE libraries discovered from constrained patterns taking into account the BK. We use the 207x11082 data set because each tag is unambiguously identified. This property is very useful to link together the information coming from several sources of BK.

The area constraint is the most meaningful constraint on the internal data for the search of such synexpression groups. On the one hand, it products large patterns (the more genes they contain, the better ; the higher the frequency is, the better). On the other hand, it enables exceptions on genes and/or biological situations contrary to the maximal patterns (Rioult et al., 2003; Blachon et al., 2007) (i.e.,

formal concepts) which require that all the connected genes are over-expressed. In domains such as gene expressions where the non-determinism is intrinsic, this lead to a fragmentation of the information embedded in the data and a huge number of patterns covering very few genes or biological situations.

We fix the area threshold thanks to statistical analysis of random datasets having the same properties as the original SAGE data. We obtain a value of 20 as an optimal area threshold to distinguish between spurious (i.e., occurring randomly) and meaningful patterns (first spurious patterns start to appear for this threshold area). Unfortunately, we get too many (several thousands) candidate patterns. Increasing the threshold of the area constraint to get a reasonable number of patterns is rather counterproductive. The constraint $area \geq 75$ led to a small but uniform set of 56 patterns that was flooded by the ribosomal proteins which generally represent the most frequent genes in the dataset. Biologists rated these patterns as valid but useless.

The most valuable synexpression groups expected by biologists have non-trivial size containing genes and situations whose characteristics can be generalized, connected, interpreted and thus transformed into knowledge. To get such patterns, constraints based on the external data have to be added to the minimal area constraint just like in the constraint *q* given in the section on integration of information sources synthetizing BK. It joins the minimal area constraint with background constraints coming from the NCBI (cf. http://www.ncbi.nlm.nih.gov) textual resources (gene summaries and adjoined PubMed abstracts). There are 46671 patterns satisfying the minimal area constraint (the part (a) of the constraint *q*), but only 9 satisfy *q*. This shows the efficiency of reduction of patterns brought by the BK. One of these patterns is of biological interest (Kléma et al.). It consists of 4 genes (KHDRBS1 NONO TOP2B FMR1) over-expressed in 6 biological situations (BM_P019 BM_P494 BM_P608 BM_P301 BM_H275 BM_H876), BM stands for brain medulloblastoma. A cross-fertilization with other external data was obviously attractive. So, we define a constraint *q'* which is similar to *q*, except that the functional Gene Ontology (cf. http://www.geneontology.org/) is used instead of NCBI textual resources. Only 2 patterns satisfy *q'*. Interestingly, the previous pattern that was identified by the expert as one of the "nuggets" provided by *q'* is also selected by *q'*. The constraints *q* and *q'* demonstrate two different ways to reach a compact and meaningful output that can be easily human surveyed.

REP summaries. Following our work to study the relationships between the genes and the type of biological situations according to cancer and no cancer, we computed REP summaries from the SAGE data. We use the same binary data set as in the characterization rules task.

Table 6 depicts the REP 4-summary with *mingr*=2. We observe that all patterns describe the class cancer. Using other values for the parameters *k* and *mingr* also leads to only characterize cancer. Interestingly, the 3 extracted genes characterize 40% of biological situations and even 61% of cancerous situations. We will see below that this REP summary confirms the results obtained with characterization rules. Nevertheless, a great interest of the approach based on the summarization is to directly isolate genes without requiring a manual inspection of rules.

A Breakthrough on Mining and Using Local Pattern Methods

A first challenge in discovery knowledge from local patterns in SAGE data is to perform the local pattern extractions. Recalling that few years ago it was impossible to mine such patterns in large datasets and only association rules with rather a high frequency threshold were used (Becquet et al., 2002). Relying on recent progress in constraint-based paradigm, we have proposed efficient data mining methods to mine local patterns solving the problem due to the size of the search space. Key ideas are the use of

Table 6. REP 4-summary of SAGE data with mingr = 2

Sequence (tag)	Description (gene)	*supp*	*gr*
cancer			
CATCCAAAAC	HNRPH1 Heterogeneous nuclear ribonucleoprotein H1 (H)	0.28	2.10
CTCTTCGAGA	GPX1 Glutathione peroxidase 1	0.32	3.28
GTTGCTGCCC	NIFIE14 Seven transmembrane domain protein	0.26	2.50

Class coverage : 40% / Running-time: 1.37s

the extension of patterns and depth-first search. Thanks to the constraint-based mining approach, the user can handle a wide spectrum of constraints expressing a viable notion of interestingness. We deal with characterization rules, emerging patterns, minimal area (which is the translation in the constraint paradigm of a synexpression group), but many other possibilities are offered to the user.

A second challenge is to deal with the (huge) collections of local patterns. We claim that we propose fruitful methods to eliminate redundancy between patterns and highlighting the most promising ones or summarizing them. Integrating BK in a data mining process is a usual work for the biologist, but he did it manually. To the best of our knowledge, there is no other constraint-based method to efficiently discover patterns from large data under a broad set of constraints linking BK distributed in various knowledge sources. Recursive mining is a new and promising way which ensures to produce very few patterns summarizing the data. These summaries can easily be inspected by the user.

Interestingness of Biological Results

A first result is that most of the extracted patterns were harboring (or even composed only of) genes encoding ribosomal proteins, and proteins involved in the translation process. This is for example the case for the vast majority of the characterization rules concluding on cancer (see Tables 4 and 5). Such an overexpression has been documented in various contexts ranging from prostate cancer (Vaarala et al., 1998) to v-erbA oncogene over-expression (Bresson et al., 2007). The biological meaning of such an over-expression is an open question which is currently investigated in the CGMC lab.

As a second lesson, we demonstrated that mining local patterns discovers promising biological knowledge. Let us come back on the pattern highlighted by the BK (see above). This pattern can be verbally characterized as follows: it consists of 4 genes that are over-expressed in 6 biological situations, it contains at most one ribosomal gene, the genes share a lot of common terms in their descriptions as well as they functionally overlap, at least 3 of the genes are known (have a non-empty record) and all of the biological situations are medulloblastomas which are very aggressive brain tumors in children. This pattern led to an interesting hypothesis regarding the role of RNA-binding activities in the generation and/or maintenance of medulloblastomas (Kléma et al.).

Finally, our data mining approaches enable a cross-fertilization of results, indicating that a relatively small number of genes keeps popping up throughout various analysis. This is typically the case of the GPX1 gene highlighted both on characterization rules and REP summaries to have an influence on the development of cancer. This gene encodes a cytosolic glutathione peroxidase acting as an antioxidant by detoxifying hydroperoxides (Brigelius-Flohe, 2006). It is known that exposition to an oxidative stress is a factor that favors development of different types of tumors (Halliwell, 2007). It is therefore

reasonable to suggest that this gene is over-expressed to respond to an oxidative stress to which cells have been exposed. It would be of interest to verify its expression level by RT-PCR in normal versus cancerous samples in human.

CONCLUSION

There are now a few methods to mine local patterns under various sets of constraints even in large data sets such as gene expression data. Nevertheless, dealing with the huge number of extracted local patterns is still a challenge due to the difficult location of the most interesting patterns. In this chapter, we have presented several methods to reduce and summarize local patterns. We have shown the potential impact of these methods on the large SAGE data. By highlighting few patterns, these approaches are precious in domains such as genomics where a manual inspection of patterns is highly time consuming. Our methods provide qualitative information (e.g., biological situations associated to genes, text resources) but also quantitative information (e.g., growth rate or other measures). Such characteristics are major features in a lot of domains with noisy data and non-deterministic phenomena for knowledge discovery. We think that our results on SAGE data illustrate the power of local patterns to highlight gene expression patterns appearing through very different conditions, and that such patterns would not be captured by global tools such as hierarchical clustering.

A future issue is the combination of these methods: how to ensure to build non redundant optimal recursive patterns? how to integrate BK in recursive mining? Another way is to design new kinds of constraints to directly mine global patterns as sets of local patterns or produce models.

ACKNOWLEDGMENT

This work has mainly been done within the Bingo project framework (http://www.info.unicaen.fr/~bruno/bingo). The work of Jiři Kléma was funded by the Czech Ministry of Education in terms of the research programme Transdisciplinary Research in the Area of Biomedical Engineering II, MSM 6840770012. The authors thank all members of the

Bingo project and especially Sylvain Blachon for generating the SAGE gene expression matrices and Jean-François Boulicaut for fruitful discussions. This work is partly supported by the ANR (French Research National Agency) funded project Bingo2 ANR-07-MDCO-014 (http://bingo2.greyc.fr/), which is a follow-up of the first Bingo project and the Czech-French PHC Barrande project “Heterogeneous Data Fusion for Genomic and Proteomic Knowledge Discovery”.

REFERENCES

Bailey, J., Manoukian, T., & Ramamohanarao, K. (2002). Fast algorithms for mining emerging patterns. *Proceedings of the Sixth European Conference on Principles Data Mining and Knowledge Discovery (PKDD’02)* (pp. 39-50). Helsinki, Finland: Springer.

Bayardo, R. J. (2005). The hows, whys, and whens of constraints in itemset and rule discovery. *Proceedings of the workshop on Inductive Databases and Constraint Based Mining* (pp. 1-13) Springer.

Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J.-F., & Gandrillon, O. (2002). Strong association rule mining for large gene expression data analysis: a case study on human SAGE data. *Genome Biology*, *3*.

Blachon, S., Pensa, R. G., Besson, J., Robardet, C., Boulicaut, J.-F., & Gandrillon, O. (2007). Clustering formal concepts to discover biologically relevant knowledge from gene expression data. *Silico Biology, 7.*

Bonchi, F., & Lucchese, C. (2006). On condensed representations of constrained frequent patterns. *Knowledge and Information Systems, 9*, 180-201.

Boulicaut, J.-F., Bykowski, A., & Rigotti, C. (2003). Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *Data Mining and Knowledge Discovery journal, 7*, 5-22. Kluwer Academics Publishers.

Bresson, C., Keime, C., Faure, C., Letrillard, Y., Barbado, M., Sanfilippo, S., Benhra, N., Gandrillon, O., & Gonin-Giraud, S. (2007). Large-scale analysis by sage reveals new mechanisms of v-erba oncogene action. *BMC Genomics, 8*.

Brigelius-Flohe, R. (2006). Glutathione peroxidases and redox-regulated transcription factors. *Biol Chem, 387*, 1329-1335.

Calders, T., Rigotti, C., & Boulicaut, J.-F. (2005). A survey on condensed representations for frequent sets. *Constraint-Based Mining and Inductive Databases* (pp. 64-80). Springer.

Crémilleux, B., & Boulicaut, J.-F. (2002). Simplest rules characterizing classes generated by delta-free sets. *Proceedings 22nd Int. Conf. on Knowledge Based Systems and Applied Artificial Intelligence* (pp. 33-46). Cambridge, UK.

De Raedt, L., Jäger, M., Lee, S. D., & Mannila, H. (2002). A theory of inductive query answering. *Proceedings of the IEEE Conference on Data Mining* (ICDM'02) (pp. 123-130). Maebashi, Japan.

De Raedt, L., & Zimmermann, A. (2007). Constraint-based pattern set mining. *Proceedings of the Seventh SIAM International Conference on Data Mining*. Minneapolis, Minnesota, USA: SIAM.

DeRisi, J., Iyer, V., & Brown, P. (1997). Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science, 278*, 680-686.

Dong, G., & Li, J. (1999). Efficient mining of emerging patterns: discovering trends and differences. *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD'99)* (pp. 43-52). San Diego, CA: ACM Press.

Halliwell, B. (2007). Biochemistry of oxidative stress. *Biochem Soc Trans, 35*, 1147-1150.

Hand, D. J. (2002*). ESF exploratory workshop on pattern detection and discovery in data mining, 2447 of Lecture Notes in Computer Science*. Chapter Pattern detection and discovery, 1-12. Springer.

Hébert, C., Blachon, S., & Crémilleux, B. (2005). Mining delta-strong characterization rules in large sage data. *ECML/PKDD'05 Discovery Challenge on gene expression data co-located with the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'05)* (pp. 90-101). Porto, Portugal.

Kléma, J., Blachon, S., Soulet, A., Crémilleux, B., & Gandrillon, O. (2008). Constraint-based knowledge discovery from sage data. *Silico Biology*, *8*(0014).

Kléma, J., & Zelezny, F. In P. Berka, J. Rauch and D. J. Zighed (Eds.), *Data mining and medical knowledge management: Cases and applications, chapter Gene Expression Data Mining Guided by Genomic Background Knowledge*. IGI Global.

Knobbe, A., & Ho, E. (2006). Pattern teams. *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'06)* (pp. 577-584). Berlin, Germany: Springer-Verlag.

Korotkina, R. N., Matskevich, G. N., Devlikanova, A. S., Vishnevskii, A. A., Kunitsyn, A. G., & Karelin, A. A. (2002). Activity of glutathione-metabolizing and antioxidant enzymes in malignant and benign tumors of human lungs. *Bulletin of Experimental Biology and Medicine, 133*, 606-608.

Li, J., Liu, G., & Wong, L. (2007). Mining statistically important equivalence classes and delta-discriminative emerging patterns. *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'07)* (pp. 430-439). New York, NY, USA: ACM.

Madeira, S. C., & Oliveira, A. L. (2004). Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Trans. Comput. *Biology Bioinform., 1*, 24-45.

Mannila, H., & Toivonen, H. (1997). Levelwise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery, 1*, 241-258.

Morik, K., Boulicaut, J.-F., & (eds.), A. S. (Eds.). (2005). *Local pattern detection, 3539* of *LNAI*. Springer-Verlag.

Nasr, M., Fedele, M., Esser, K., & A, D. (2004). GPx-1 modulates akt and p70s6k phosphorylation and gadd45 levels in mcf-7 cells. *Free Radical Biology and Medicine, 37*, 187-195.

Ng, R. T., Lakshmanan, V. S., Han, J., & Pang, A. (1998). Exploratory mining and pruning optimizations of constrained associations rules. *Proceedings of ACM SIGMOD'98* (pp. 13-24). ACM Press.

Pan, F., Cong, G., Tung, A. K. H., Yang, Y., & Zaki, M. J. (2003). CARPENTER: finding closed patterns in long biological datasets. *Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'03)* (pp. 637-642). Washington, DC, USA: ACM Press.

Pensa, R., Robardet, C., & Boulicaut, J.-F. (2005). A bi-clustering framework for categorical data. *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'05)* (pp. 643-650). Porto, Portugal.

Rioult, F., Boulicaut, J.-F., Crémilleux, B., & J., B. (2003). Using transposition for pattern discovery from microarray data. *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'03)* (pp. 73-79). San Diego, CA.

Siebes, A., Vreeken, J., & Van Leeuwen, M. (2006). Item sets that compress. *Proceedings of the Sixth SIAM International Conference on Data Mining*. Bethesda, MD, USA: SIAM.

Soulet, A. (2007). Résumer les contrastes par l'extraction récursive de motifs. Conférence *sur l'Apprentissage Automatique (CAp'07)* (pp. 339-354). Grenoble, France: Cépaduès Edition.

Soulet, A., & Crémilleux, B. (2005). An efficient framework for mining flexible constraints *Proceedings 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'05)* (pp. 661-671). Hanoi, Vietnam: Springer.

Soulet, A., & Crémilleux, B. (2008). Soulet A., Crémilleux B. Mining constraint-based patterns using automatic relaxation. *Intelligent Data Analysis, 13*(1). IOS Press. To appear.

Soulet, A., Crémilleux, B., & Rioult, F. (2004). Condensed representation of emerging patterns. *Proceedings 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'04)* (pp. 127-132). Sydney, Australia: Springer-Verlag.

Soulet, A., Kléma, J., & Crémilleux, B. (2007). *Post-proceedings of the 5th international workshop on knowledge discovery in inductive databases in conjunction with ECML/PKDD 2006 (KDID'06), 4747 of Lecture Notes in Computer Science, chapter Efficient Mining under Rich Constraints Derived from Various Datasets*, 223-239. Springer.

Vaarala, M. H., Porvari, K. S., Kyllonen, A. P., Mustonen, M. V., Lukkarinen, O., & Vihko, P. T. (1998). Several genes encoding ribosomal proteins are over-expressed in prostate-cancer cell lines: confirmation of l7a and l37 over-expression in prostate-cancer tissue samples. *Int. J. Cancer, 78*, 27-32.

Velculescu, V., Zhang, L., Vogelstein, B., & Kinzler, K. (1995). Serial analysis of gene expression. *Science, 270*, 484-487.

KEY TERMS

Background Knowledge: Information sources or knowledge available on the domain (e.g., relational and literature databases, biological ontologies).

Constraint: Pattern restriction defining the focus of search. It expresses a potential interest given by a user.

Functional Genomics: Functional Genomics hints at understanding the function of genes and other parts of the genome.

Local Patterns: Regularities that hold for a particular part of the data. It is often required that local patterns are also characterized by high deviations from a global model (Hand, 2002).

Recursive Mining: Repeating the pattern mining process on output to reduce it until few and relevant patterns are obtained.

SAGE Method: SAGE produces a digital version of the thanscriptome that is made from small sequences derived from genes called "tags" together with their frequency in a given biological situation.

Chapter 3

Set-level Microarray Classification

Cross-genome knowledge-based expression data fusion

Matěj Holec, Jiří Kléma, Filip Železný, Jiří Bělohradský
Czech Technical University,
Technická 2, Prague 6, 166 27
{holecm1,klema,zelezny}@fel.cvut.cz

Jakub Tolar
University of Minnesota, Minneapolis
tolar003@umn.edu

Abstract

This paper presents the web tool XGENE.ORG which facilitates the integration of gene expression measurements with background genomic information, in particular the gene ontology and KEGG pathways. The novelty of the proposed data fusion is in the introduction of working units at different levels of generality acting as sample features, replacing the commonly used gene units, consequently allowing for cross-genome (multi-platform) expression data analysis. The integration of different microarray platforms contributes to the robustness of knowledge extracted when single-platform samples are rare and facilitates inference of biological knowledge not constrained to single organisms.

1. Introduction

In the current post-genomic era, various aspects of gene functions are being uncovered by a large number of experiments producing huge amounts of heterogeneous data at an accelerating pace. Putting all this data together, while taking into account existing knowledge has become a pressing need for developing tools able to explore and simulate biological entities at a system level. A popular example is the microarray (MA) technology enabling to simultaneously estimate the activity of tens of thousands genes (virtually the entire genome) in a sample tissue. Early research studies exploited gene expression data to discover sets of marker probesets, e.g. those with elevated expression in a cancerous tissue. Despite several successes in predictive diagnosis using such obtained knowledge, it is now generally agreed that the true logic of diseases and other biological processes can only be explained by detailed interpretation of the measurements, clarifying how and why certain genes follow certain expression patterns in certain situations. This in turn requires to integrate the large volumes of raw measurements with another huge body of available additional information (or background knowledge, BK), such as known gene functions, mutual interactions or roles in regulatory and signalling pathways.

The most popular and frequent utilization of background knowledge is based on enrichment analysis. The state-of-the-art tools such as DAVID [7] search for enriched apriori-defined *gene groups*, rather than interpret individual differentially expressed probesets (or genes[1]). The principal foundation of enrichment analysis is that if a biological process is abnormal, the co-functioning genes should have a higher (enriched) potential to be selected as relevant. Such a rationale can move the analysis from an individual gene-oriented to a relevant gene group-based one. The overview of 68 available enrichment tools is available in [8]. The biological utility of pathways was demonstrated by the study [11] where a significantly downregulated pathway-based gene set in a class of type 2 diabetes was discovered despite no single significant gene being detected. [10] provides a method that uses gene ontology terms and their grouping to improve the interpretation of gene set enrichment for microarray data.

This paper presents the web tool XGENE.ORG available at `http://xgene.org`. Similarly to enrichment tools, XGENE.ORG tool facilitates integration of large volumes of raw gene expression measurements with another huge body of available genomic information. Contrary to existing enrichment tools, it offers additional functionality resulting from a data-fusion strategy based apriori defined gene sets. In particular, the main resulting feature of the present tool is that it enables to analyze gene expression data collected from heterogeneous platforms in an integrated manner. The heterogeneous platforms may pertain to different organism species. The significance of this contribution is at least twofold. First, microarray experiments are costly, often resulting in numbers of samples insufficient for reliable modeling. The possibility of systematically integrating the experimenter's data with numerous public expression samples coming from heterogeneous platforms, would obviously help the experimenter. Second, such in-

[1] In this paper we consider probesets and genes as closely related but still distinct units as several probesets may interrogate the same gene.

tegrated analysis provides the principal means to discover biological markers shared by different-genome species.

XGENE.ORG explicitly implements various *working units* and determines their *level of activity*. The activity of a superior (more abstract) working unit is calculated from the known (measured) activity of a set of inferior (less general) working units. For example, it selects all the probesets that are annotated by the same gene identifier and computes gene activity. Likewise, all the genes whose products act in a single pathway are used to compute pathway activity. A similar approach that applies a method based on singular value decomposition to calculate pathway activity was proposed in [19]. However, XGENE.ORG takes a step forward. First, it works with a various types of working units on different levels of generality. Second, it uses them to perform cross-genome and cross-organism analysis as there are working units that generalize beyond individual platforms and species. Third, in addition to standard statistical analyses, it applies machine learning (ML) techniques to develop interpretable models that distinguish among user-defined classes.

Let us exemplify some of the currently available types of working units. The first type that enables cross-platform analysis aggregates measurements that share a common gene ontology (GO) [3] term. The second type aggregates measurement units acting in the same biological pathways formalized by the KEGG [9] database. The third type represents a further novel contribution of our work and is based on the notion of a *fully coupled flux*, which is a pattern prescribing pathway partitions hypothesized by [12] to involve strongly co-expressed genes.

To sum up, analyses and models based solely on *measurement units* defined by the individual probesets whose expression is immediately measured by microarrays suffer from the inherent microarray noise and often fail to identify subtle patterns, give a large room to overfitting and prove hard to interpret and apply. Genomic background knowledge makes it possible to introduce and analyze alternative working units that avoid the bottlenecks mentioned above and provide improved interpretation power and statistical significance of analysis results. At the same time, different platforms and/or species deal with different sets of measurement units that cannot be directly matched. Consequently, multi-platform analyses cannot be performed without working units whose meaning is general enough to be defined in each platform and whose activity can unambiguously be evaluated in each sample independently of its platform type. Working units then serve as markers (or features) to distinguish between user-supplied sample classes.

The paper is organized as follows. In Section 2 we synthetize the system's functionality and describe its architecture. Section 3 describes the methodological elements of our approach, consisting of normalization, extraction of working units at various levels of generality, testing their significance, and predictive classification. Section 4 briefly exemplifies the use of the system through two case studies. Section 5 lays out prospects for future work and concludes the paper.

2. System Description

The main goal of the presented XGENE.ORG tool is to analyse a wide range of publicly accessible heterogeneous gene expression samples. The tool provides an interface to search available measurements whose annotation is relevant to the studied biological topic. Typically, a set of relevant measurements straddles various microarray platforms and organisms. There are two principal reasons to allow for their integration. The technical reason concerns the sufficiency of sample sets for reliable modeling. The more platforms accessed, the larger number of samples is at hand. The scientific reason pertains to the relevance of the outcomes. Combining multi-platform input data contributes to the generality of any knowledge discovered.

The tool operates in three basic phases:

1. define sample classes of interest; search and collect existing measurements representing these classes,
2. compute the activation levels of various working units with respect to the collected samples,
3. apply statistical, machine learning and visualization methods to obtain models distinguishing between the defined classes, with the pre-computed activity levels of working units acting as sample features.

XGENE.ORG implements this workflow, facilitating all three phases above. The architecture of the tool is depicted in Figure 1. XGENE.ORG integrates data from several publicly accessible databases.

Regarding the first phase above, our tool provides an interface to the Gene Expression Omnibus (GEO) [1]. XGENE.ORG enables a keyword-based search and filtering of individual gene expression measurements as illustrated in Fig 2. GEO is currently the largest public repository archiving and freely distributing high-throughput gene expression measurement data submitted by the scientific community. GEO currently stores approximately a billion individual gene expression measurements, derived from over 100 organisms, addressing a wide range of biological issues. GEO is accessible at www.ncbi.nlm.nih.gov/geo. The interaction with GEO is supervised by the user. The measurements are normalized and saved in the internal PROLOG format that simplifies subsequent integration of the expression data with data capturing biological process structure (pathways) and relational information (the gene ontology).

Secondly, XGENE.ORG accesses the databases that provide background knowledge required to define and interpret

the predefined set of working unit types (they are discussed in detail thereunder). The individual microarray platforms are annotated by the Bioconductor packages [4]. Bioconductor packages also provide annotations by the gene ontology terms. The background knowledge on pathways and fluxes is taken directly from KEGG [9] database. The background knowledge management is fully automated and carried out without user interventions. The tool downloads all the packages and datasets needed to analyse the measurements currently selected by the user and stores them in the internal PROLOG representation.

The critical step is to fuse the collected measurements and background knowledge into unified cross-platform data subsequently accessed by the statistical and machine learning tools. Within this fusion, working units are computed across samples taken from various platforms and organisms. The resulting unified representation consists of a single matrix in which rows correspond to samples, columns correspond to working units and the respective matrix cells express the activity of a given unit within a given sample as a real value. Each working unit subsequently serves as a statistical variable for tasks such as fold change analysis, or a *sample feature* for machine learning algorithms.

Currently, three kinds of analysis results are supported:

- a classifier that estimates the sample class given an expression sample and its platform label
- a list of working units significantly differentially expressed in classes
- a scatterplot that shows class distribution in a (transformed 2D) space of working units.

The results are provided to the user in the form of hypertext, including links pointing to detailed descriptions working units employed in the displayed result.

The interaction with the user who starts a new experiment consists of the following steps:

1. The user logs to his/her personal account. This account stores the user's previous experiments and their results.
2. The user creates a new experiment. The experiment can be entirely new (the interaction proceeds by the following step) or it can be derived from a previous experiment (the experiment then inherits the classes and datasets defined earlier and thus skips the two following steps).
3. The user creates and entitles two or more of sample classes. These classes contain no measurement samples at this stage.
4. The user fills each of the defined classes with a set of relevant GEO expression samples. The samples are preselected via keyword-based search and then finely filtered by the user on the basis of experimental annotations (see Figure 2),

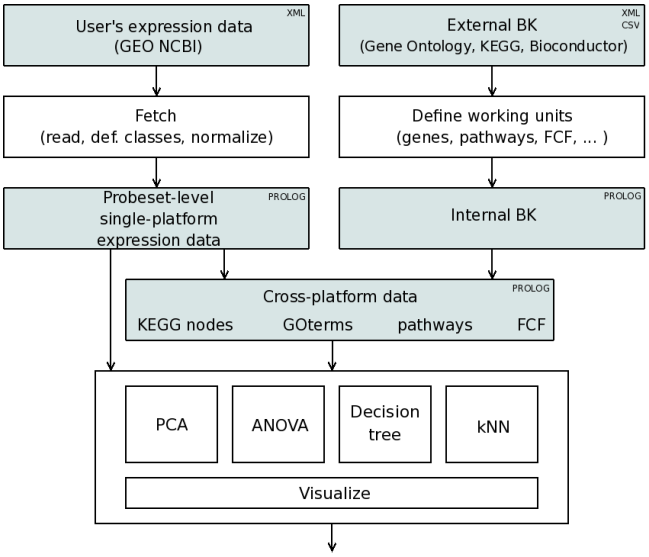


Figure 1. XGENE.ORG architecture

5. The user selects (possibly repeatedly) proper working units, platform types and algorithms and starts the experiment.
6. The system collects the necessary background knowledge, computes the working units defined above and applies the selected algorithms.
7. The computation begins and the user can log out. (S)he is informed by email as soon as the results are ready to be shown.
8. The user views the results. A result-filter helps user's orientation if a large number of result types has been requested in step 5.

3. Methods

This section describes the methodological elements of our approach. It gives an overview of working units and shows the way in which their activity is estinated and evaluated. It specifies the statistical methods serving to identify differentially expressed working units. It also gives a summary of currently implemented machine learning methods. Their application is at least twofold. The first one is practical. They provide means to distinguish among sample classes when the sample annotation is unknown. The second one is exploratory. As one of the keynotes of XGENE.ORG is to prove applicability of cross-platform working units, the *classification accuracy* of machine learning models is instrumental for relevance assessment of a given set of working units.

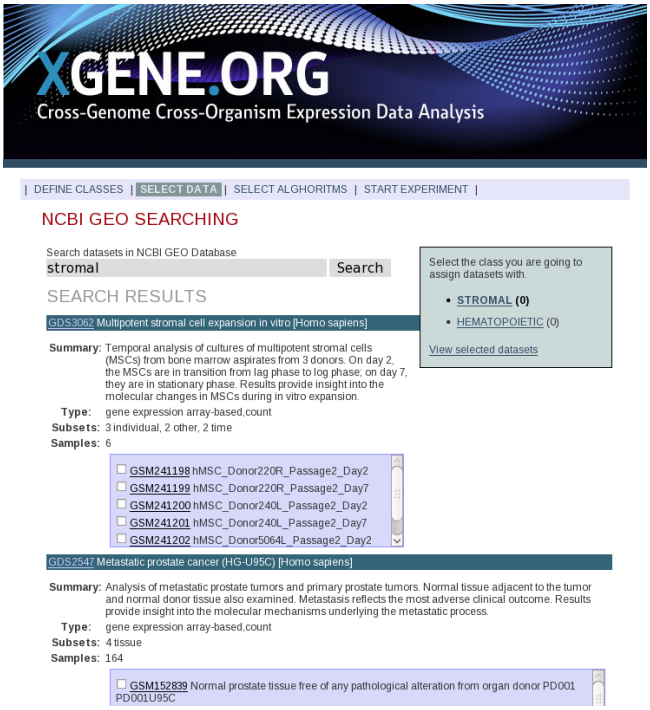


Figure 2. XGENE.ORG: collecting relevant samples from NCBI GEO. Clicking on a sample identifier ('GSMxxxxx') opens a detailed description of that sample.

a pathway node makes up the *pathway node* working unit. To compute the activity of a working unit, the probesets that transitively link to that working unit are considered. For example, the activity of a pathway is computed by aggregating the activity of all probesets corresponding to genes which in turn correspond to nodes contained in the given pathway. Obviously, this mapping is platform dependent; pathways have different probeset interpretations in different platforms. At the same time, this mapping is organism dependent and thus we have to deal with organism orthologs of pathways.

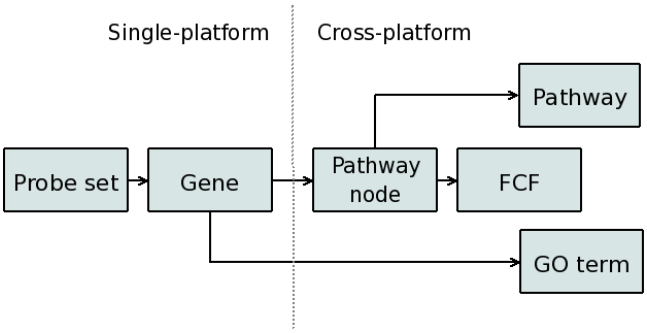


Figure 3. The hierarchy of working units. An arrow from X to Y denotes that unit Y refers to a set of X units. This relation is transitive and thus all units can ultimately be represented as families of probesets.

3.1. Working units – types and activity

Currently, we consider two principal knowledge sources in order to define working units—the gene ontology database [3] and the KEGG database [9]. The Bioconductor annotation packages [4] serve to translate among the identifiers used by the microarray manufacturers (currently, only Affymetrix is supported), and the two mentioned background knowledge databases. The widely spread EntrezIds (gene identifiers) introduced by NCBI play the role of intermediate translation identifiers. The current hierarchy of working units as implemented in XGENE.ORG is shown in Figure 3. The ultimate working units correspond to the measurement units, i.e., the probesets. Their activity in the individual samples is directly reported in the GEO input files. A single GEO file corresponds to a single microarray sample, a whole sample is represented by a probeset activity vector. The set of measured probesets is platform dependent, i.e., the vectors taken form different platforms cannot be directly matched. The more general units are gradually inferred from their subordinate units. For example, the list of probesets that are annotated by the same gene identifier makes up the *gene* working unit. The list of genes linked to

Significance testing at the level of pathways and/or GO terms is a standard method widely implemented in enrichment tools. However, these working units may prove overly general to capture subtle biological dependencies. Many notable biological conditions are characterized by the activation of only certain parts of pathways; for example, see references [16, 20, 18]. The notion of 'pathway activation' implied by the notion of pathway working units may thus violate intuition and hinder interpretation. Therefore we also extracted all pathway partitions which comply with the graph-theoretic notion of fully coupled flux [12]. It is known that the genes coupled by their enzymatic fluxes not only show similar expression patterns, but also share transcriptional regulators and frequently reside in the same operon in prokaryotes or similar eukaryotic multigene units such as the hematopoietic globin gene cluster. FCF is a special kind of network flux that corresponds to a pathway partition in which non-zero flux for one reaction implies a non-zero flux for the other reactions and vice versa. It is the strongest qualitative connectivity that can be identified in a network. The notion of an FCF is explained through an example in Fig. 4; for a detailed definition, see reference [12]. Again, a probeset falls in a list corre-

sponding to a FCF if it is mapped to a KEGG node in some organism-ortholog of that FCF. To conclude, XGENE.ORG uses working units at various levels of generality. This hierarchy of units allows to capture and interpret biological issues that most strongly manifest in various kinds of existing biological models.

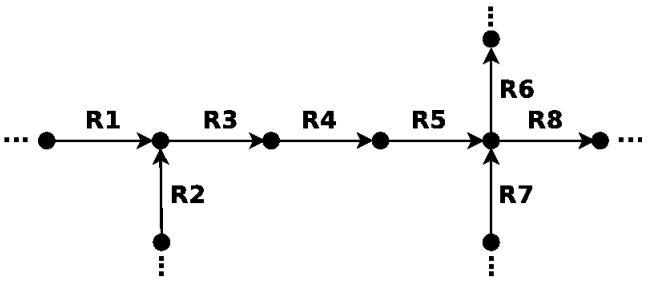


Figure 4. Fully coupled fluxes in a simplified network with nodes representing chemical compounds and arrows as symbols for chemical reactions among them. Each arrow can be labeled by a protein. R3, R4 and R5 are fully coupled as a flux in any of these reactions implies a flux in the rest of them. Note that R1 and R3 do not constitute a FCF as a flux in R3 does not imply a flux in R1.

The extraction of working units and computation of their activity in biological samples was conducted in Prolog. The process of computation of KEGG node activity in a sample set that originates from two different platforms is shown in Figure 5.

Currently, the aggregated activity of a unit in a sample is computed as the mean activity of all the measurement units that map on it in the given platform. When averaging is applied in Figure 5, it holds that $k_{wi} = (p_{xi} + p_{yi})/2 = g_{zi}$ and $k_{wj} = (p'_{aj} + p'_{bj} + p'_{cj})/3 = (g'_{dj} + 2g'_{ej})/3$. It means that the weight of gene g'_e is twofold with respect to g'_d as the former maps to two probesets while the latter to one probeset only. We are aware that averaging is an elementary aproach that may oversimplify the relationships and information transmission among units. Finding a biologically sound way to model the activity of genomic entities from microarray data is an open complex research issue. First of all, the mapping between probesets and genes is not unambiguous because the individual probesets map to more than one transcript dependent upon the biological condition [17]. There are efforts to refine the standard annotation of microarray probesets from gene level to transcript and protein level [22]. Secondly, it is advantageous to take into account internal structure of the modelled entities. More profound knowledge-based approaches to generalize towards more complex entities such as pathways can be found in [13, 15, 14]. However, these works always focus at a single type of applied knowledge and do not concern a universal workflow with multiple platforms on its input. Moreover, the application of such more sophisticated strategies to aggregate statistical values pertaining to subunits to represent analogical values of more general units is not scalable in the framework adopted by XGENE.ORG. In principle, this is because the simple average computation among subunits would have to be replaced by some sort of *subset selection*. Here, one searches for the best subset of subunits that best represent the parent unit, according to some optimality criterion. Generally, searching among subsets in a family of probesets S becomes quickly intractable with the growing size of S. For example, a selection of the best family of probesets for a given *gene* may be tractable as there is typically just a few probesets mapping to a gene in a platform. However, searching among subsets among in the pool of 10s-100s of probesets mapping to a *pathway* is generally no longer tractable.

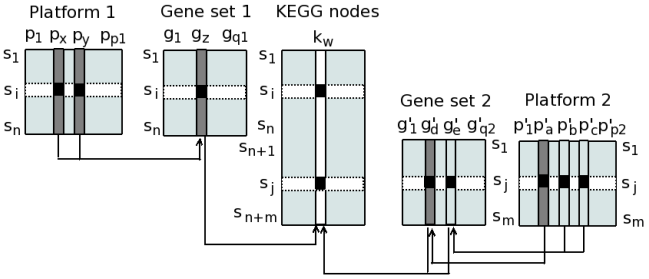


Figure 5. KEGG node activity. The activity of the node k_w in the sample s_i denoted as k_{wi} is given by the activity of its subordinate gene g_z whose activity is in turn given by the activity of its subordinate probesets p_x and p_y measured in Platform 1. The activity of the same node k_w in the sample s_j denoted as k_{wj} is given by the activity of g'_d and g'_e. The activity of g'_d is given by the activity of p'_a while activity of g'_e is inferred from activity of p'_b and p'_c measured in Platform 2.

3.2. Analysis Algorithms

After the collection of all data needed for a defined experiment, *normalization* is conducted separately for each involved platform to consolidate same-platform samples. Quantile normalization [2] ensures that the distribution of expression values across such samples is identical. As a second step, scaling provides means to consolidate the measurements across multi-platform samples. We subtract the

sample mean from all sample components, and divide them by the standard deviation within the sample. As a result, all samples independently of the platform exhibit zero mean and unit variance. We conduct these steps using the Bioconductor [4] software.

After normalization, the most basic type of analysis that may be generated on user's request is *fold change* analysis whose goal is to rank the ability of the individual working units to distinguish among the user-defined classes. For this sake, we apply the one-way ANOVA (analysis of variance) method. In single platform tests where ANOVA ranks probesets, it determines if the sample distribution among classes has a significant effect on probe-set expression behavior. A significant p-value resulting from a one-way ANOVA test indicates that a probeset is differentially expressed in at least one of the classes analyzed. The lower p-value, the higher the probeset ranking. When ranking units of higher order, we do not proceed in a post hoc fashion from the single p-values of probe-sets but we model the expression of working units directly. For every unit, a complete list of probesets that map onto that unit is taken independently of the platform type. Their expression values in all the samples are gathered and factorized by the user-defined class variable[2]. With such prepared data, one-way ANOVA is run. Using the distinction for gene set statistical testing carried out in [5] we apply a self-contained test with subject sampling. No averaging is applied.

Having a single-tabular representation in which activity of a set of working units is computed across samples, a wide-scale of machine learning algorithms can be applied. The most interesting appear to be such algorithms that allow for direct human interpretation of the resulting models and still keep a good predictive power. Specifically, we included the J48 decision tree learner provided by the machine learning environment WEKA [21]. The K-nearest neighbor (kNN) algorithm from the same environment has also been included.

Finally, principal component analysis (PCA) is used for the purpose of dimensionality reduction in a space of working units with subsequent visualisation of samples [6]. PCA is known to retain those characteristics of the data set that contribute most to its variance. In XGENE.ORG it helps to exhibit class distribution in 2D and visually assess the potential of a set of working units to distinguish among classes.

4. Case studies

Here we demonstrate our methodology in two biological case studies. We address general tasks of tissue type classification. The first experiment focuses on distinct features of blood-forming (*hematopoietic*) and supportive (*stromal*) cellular compartments in the bone marrow. The second assesses differences in brain, liver and muscle tissues. Both experiments are of biological significance as they tackle novel challenges in understanding of cellular behavior: the former in the complex functional unit termed hematopoietic stem cell niche, where inter-dependent hematopoietic and stromal cell functions synergize in the blood-forming function of the bone marrow; the latter in comparison of cell fate determined by the tissue origin from the separate layers of the embryo: ectoderm (brain), endoderm (liver) and mesoderm (muscle). While of general character, the chosen tasks are not just random biological exercises as these studies may illuminate cellular functions determined by gene expression signatures in complex cell system seeded by cell-type-heterogeneous undifferentiated populations (hematopoietic and stromal stem cells in the cell niche), and in the cell-type-homogeneous differentiated tissues (brain, liver and muscle), respectively.

The significance tests at gene level identified elevated expression of genes canonical for the specific tissue studied, such as myelin basic protein in brain, isocitrate dehydrogenase in liver, tropomyosin in muscle and differential expression of integrin beta 5 inhematopoietic and stromal cell populations of the bone marrow.

The experiments with machine learning algorithms proved that working units applicable across platforms clearly distinguish among classes in both studies. The resulting models are compact, easy to interpret and accurate. Fig. 6 exemplifies the application of the decision tree learner J48 on the level of FCFs in the brain/liver/muscle study. The model tested by 10-fold cross-validation reaches the classification accuracy nearly 98%, it misclassifies 3 out of 131 samples. The tree has only 2 internal nodes (2 activity tests that put into use two FCFs) and 3 leaves (one leaf per class).

A similar conclusion follows from PCA visualizations (Fig. 7). The activity of working units tends to share the same pattern within classes as well as within the same platforms or the same laboratories. However, the class pattern is strong enough to clearly distinguish among classes independently of platform.

The complete overview of results is available via the XGENE.ORG webpage.

5. Discussion

XGENE.ORG is a web tool for analysis of gene expression data collected from heterogeneous (multi-platform) microarray platforms under the presence of genomic background knowledge. The integration of multi-platform data is conducted automatically by using the available genomic

[2] In Figure 5, the significance of k_w is inferred from concatenation of the expression vectors for probesets p_x, p_b, p'_a, p'_b and p'_c. The factorization is given by the sample distribution which is not shown.

background knowledge to define candidate working units general enough to be quantified in any sample regardless of the platform on which it was measured. The heterogeneous data are transformed into a single-tabular representation which summarizes the activity of the working units for all the collected samples. Such a unified representation lends itself to various types of analysis provided by XGENE.ORG based on statistical or machine learning methods.

The contribution of this tool is at least twofold. First, microarray experiments are costly, often resulting in numbers of samples insufficient for reliable modeling. The possibility of systematically integrating the experimenter's data with numerous public expression samples coming from heterogeneous platforms, would obviously help the experimenter. Second, such integrated analysis provides the principal means to discover biological markers shared by different-genome species.

Acknowledgments. The XGENE.ORG server code adopts parts of the open-source software Bioconductor [4] and WEKA [21]. For displaying pathway maps (such as in Fig. 7), XGENE.ORG uses data provided by the KEGG database [9]. The Czech-US coordination and travel of FZ is funded by the Czech Ministry of Education through project ME910. The authors are supported by the Czech Grant Agency through project 201/09/1665 (MH), the Czech Ministry of Education through projects MSM6840770038 (FZ) and MSM6840770012 (JK), and by the Children's Cancer Research Fund of the University of Minnesota (JT).

References

[1] T. Barrett, D. B. Troup, S. E. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. F. Kim, A. Soboleva, M. Tomashevsky, and R. Edgar. Ncbi geo: mining tens of millions of expression profiles–database and tools update. *Nucleic Acids Res*, 35(Database issue), January 2007.

[2] B. M. Bolstad, R. A. Irizarry, M. Astrand, and T. P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, January 2003.

[3] T. G. O. Consortium. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25, 2000.

[4] R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, K. Hornik, T. Hothorn, W. Huber, S. Iacus, R. Irizarry, F. Leisch, C. Li, M. Maechler, A. J. Rossini, G. Sawitzki, C. Smith, G. Smyth, L. Tierney, J. Y. H. Yang, and J. Zhang. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80, 2004.

[5] J. Goeman and P. Bühlmann. Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics*, 23(8):980–987, 2007.

[6] N. S. Holter, M. Mitra, A. Maritan, M. Cieplak, J. R. Banavar, and N. V. Fedoroff. Fundamental patterns underlying gene expression profiles: Simplicity from complexity. *Proceedings of the National Academy of Sciences*, 97(15):8409–8414, July 2000.

[7] D. W. Huang, B. T. Sherman, and R. A. Lempick. Systematic and integrative analysis of large gene lists using david bioinformatics resources. *Nature Protocols*, 4:44 – 57, 2009.

[8] D. W. W. Huang, B. T. T. Sherman, and R. A. A. Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic acids research*, November 2008.

[9] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic Acids Res*, 32:277–280, 2004.

[10] A. Lewin and I. C. Grieve. Grouping gene ontology terms to improve the assessment of gene set enrichment in microarray data. *BMC Bioinformatics*, 7:426+, October 2006.

[11] V. Mootha, C. Lindgren, and S. L. et al. Pgc-1-alpha-responsive genes involved in oxidative phosphorylation are coorinately down regulated in human diabetes. *Nature Genetics*, 34:267–273, 2003.

[12] R. A. Notebaart, B. Teusink, R. J. Siezen, and B. Papp. Co-regulation of metabolic genes is better explained by flux coupling than by network distance. *PLOS Computational Biology*, 4(1), 2008.

[13] J. Rahnenführer, F. S. Domingues, J. Maydt, and T. Lengauer. Calculating the statistical significance of changes in pathway activity from gene expression data. *Stat Appl Genet Mol Biol*, 3, 2004.

[14] F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J.-P. Vert. Classification of microarray data using gene networks. *BMC Bioinformatics*, 8:35+, February 2007.

[15] G. Schramm, M. Zapatka, R. Eils, and R. König. Using gene expression data and network topology to detect substantial pathways, clusters and switches during oxygen deprivation of escherichia coli. *BMC Bioinformatics*, 8:149, 2007.

[16] A. Shaw and E. Filbert. Scaffold proteins and immune-cell signalling. *Nat Rev Immunol.*, 9(1):47–56, 2009.

[17] M. A. Stalteri and A. P. Harrison. Interpretation of multiple probe sets mapping to the same gene in affymetrix genechips. *BMC Bioinformatics*, 8:13+, January 2007.

[18] Y. Y. Sun and J. Chen. mTOR signaling: PLD takes center stage. *Cell Cycle*, 7(20):3118–23, 2008.

[19] J. Tomfohr, J. Lu, and T. B. Kepler. Pathway level analysis of gene expression using singular value decomposition. *BMC Bioinformatics*, 6, 2005.

[20] T. Weichhart and M. Semann. The PI3K/Akt/mTOR pathway in innate immune cells: emerging therapeutic applications. *Ann Rheum Dis.*, Suppl 3:iii:70–4, 2008.

[21] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann, San Francisco, 2005.

[22] H. Yu, F. Wang, K. Tu, L. Xie, Y.-Y. Li, and Y.-X. Li. Transcript-level annotation of affymetrix probesets improves the interpretation of gene expression data. *BMC Bioinformatics*, 8:194+, June 2007.

```
J48 pruned tree
------------------

FCF592 <= -0.1778: muscle (62.0)
FCF592 > -0.1778
|   FCF81 <= -0.2503: brain (53.0)
|   FCF81 > -0.2503: liver (19.0)

Number of Leaves  :     3
Size of the tree :      5

=== Stratified cross-validation ===

Correctly Classified Instances         131               97.7612 %
Incorrectly Classified Instances         3                2.2388 %

=== Confusion Matrix ===

  a  b  c   <-- classified as
 61  0  1 |  a = muscle
  0 18  1 |  b = liver
  0  1 52 |  c = brain
```

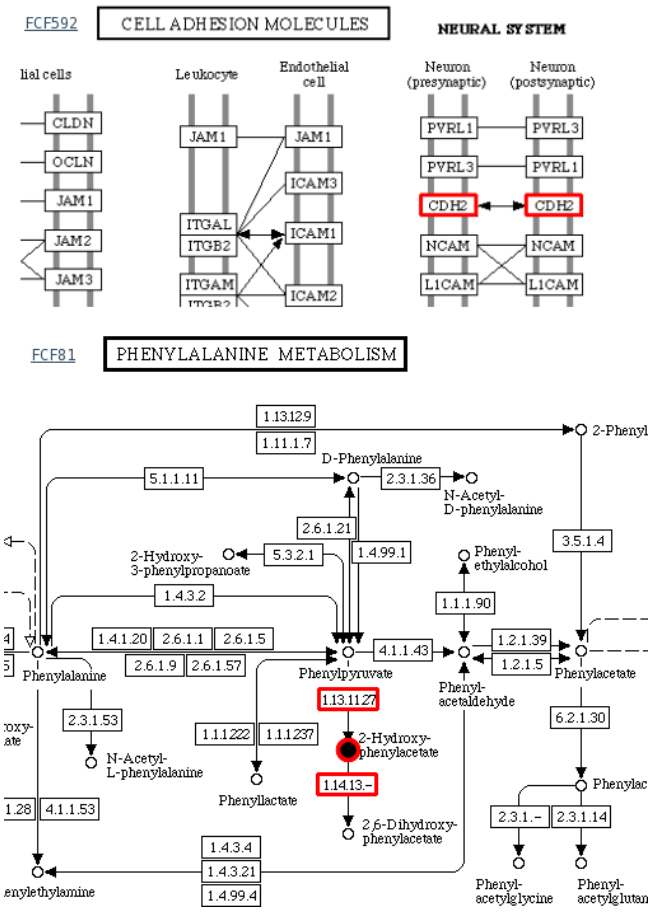


Figure 6. The flux-based cross-platform decision tree for the brain/liver/muscle study. The tree is very compact, the class is determined by two activity thresholds on two fluxes, the fluxes are visualized using KEGG pathway maps (in bold).

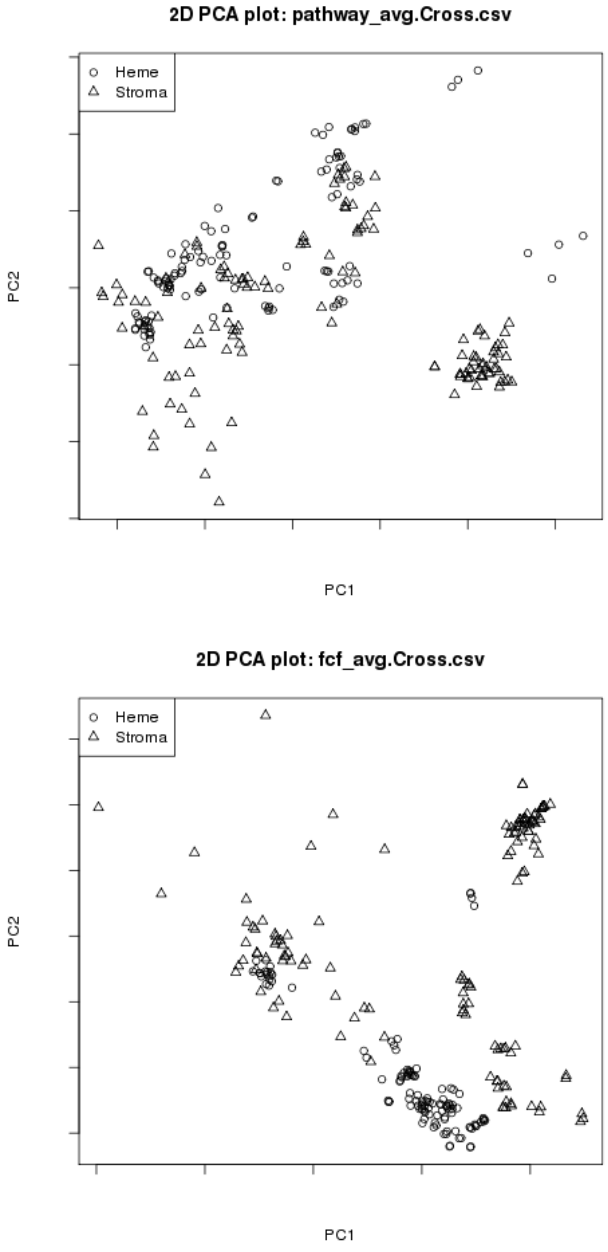


Figure 7. PCA in the hematopoietic/stromal study. The first subfigure shows cross-platform PCA in the space of pathways, the second subfigure uses FCFs instead. FCFs seem to better separate the classes (which is also confirmed by a higher classification accuracy if FCFs are used).

Holec et al. BMC Bioinformatics 2012, 13(Suppl 10):S15
http://www.biomedcentral.com/1471-2105-13-S10-S15

BMC Bioinformatics

PROCEEDINGS Open Access

Comparative evaluation of set-level techniques in predictive classification of gene expression samples

Matěj Holec[1], Jiří Kléma[1*], Filip Železný[1], Jakub Tolar[2]

From 7th International Symposium on Bioinformatics Research and Applications (ISBRA'11)
Changsha, China. 27-29 May 2011

Abstract

Background: Analysis of gene expression data in terms of a priori-defined gene sets has recently received significant attention as this approach typically yields more compact and interpretable results than those produced by traditional methods that rely on individual genes. The set-level strategy can also be adopted with similar benefits in predictive classification tasks accomplished with machine learning algorithms. Initial studies into the predictive performance of set-level classifiers have yielded rather controversial results. The goal of this study is to provide a more conclusive evaluation by testing various components of the set-level framework within a large collection of machine learning experiments.

Results: Genuine curated gene sets constitute better features for classification than sets assembled without biological relevance. For identifying the best gene sets for classification, the Global test outperforms the gene-set methods GSEA and SAM-GS as well as two generic feature selection methods. To aggregate expressions of genes into a feature value, the singular value decomposition (SVD) method as well as the SetSig technique improve on simple arithmetic averaging. Set-level classifiers learned with 10 features constituted by the Global test slightly outperform baseline gene-level classifiers learned with all original data features although they are slightly less accurate than gene-level classifiers learned with a prior feature-selection step.

Conclusion: Set-level classifiers do not boost predictive accuracy, however, they do achieve competitive accuracy if learned with the right combination of ingredients.

Availability: Open-source, publicly available software was used for classifier learning and testing. The gene expression datasets and the gene set database used are also publicly available. The full tabulation of experimental results is available at http://ida.felk.cvut.cz/CESLT.

Background

Set-level techniques have recently attracted significant attention in the area of gene expression data analysis [1-7]. Whereas in traditional analysis approaches one typically seeks individual genes differentially expressed across sample classes (e.g. cancerous vs. control), in the set-level approach one aims to identify entire sets of genes that are significant, e.g. in the sense that they contain an unexpectedly large number of differentially expressed genes. The gene sets considered for significance testing are defined prior to analysis, using appropriate biological background knowledge. For example, a defined gene set may contain genes acting in a given cellular pathway or annotated by a specific term of the gene ontology. The main advantage brought by set-level analysis is the compactness and improved interpretability of analysis results due to the smaller number of the set-level units compared to the number of genes, and more background knowledge available to such units. Indeed, the long lists of differentially expressed genes characteristic of

* Correspondence: klema@fel.cvut.cz
[1]Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, 166 27, Czech Republic
Full list of author information is available at the end of the article

traditional expression analysis are replaced by shorter lists of more informative units corresponding to actual biological processes.

Predictive classification [8] is a form of data analysis going beyond the mere identification of differentially expressed units. Here, units deemed significant for the discrimination between sample classes are assembled into formal models prescribing how to classify new samples that contain yet unknown class labels. Predictive classification techniques are thus especially relevant to diagnostic tasks and as such have been explored since very early studies on microarray data analysis [9]. Predictive models are usually constructed by supervised machine learning algorithms [8,10] that automatically discover patterns among samples with already available labels (so-called *training samples*). Learned classifiers may take diverse forms ranging from geometrically conceived models such as *Support Vector Machines* [11], which have been especially popular in the gene expression domain, to symbolic models such as logical rules or decision trees that have also been applied in this area [12-14].

The combination of set-level techniques with predictive classification has been suggested [7,15,16] or applied in specific ways [4,17-20] in previous studies, however, a focused exploration of the strategy has commenced only recently [21,22].

The set-level framework is adopted in predictive classification as follows. Sample features originally bearing the (normalized) expressions of individual genes are replaced by features corresponding to gene sets. Each such feature aggregates the expressions of the genes contained in the corresponding set into a single real value; in the simplest case, it may be the average expression of the contained genes. The expression samples are then presented to the learning algorithm in terms of these derived, set-level features. The main motivation for extending the set-level framework to the machine learning setting is again the interpretability of results. Informally, classifiers learned using set-level features acquire forms such as "predict cancer if pathway P1 is active and pathway P2 is not" (where *activity* refers to aggregated expressions of the member genes). In contrast, classifiers learned in the standard setting derive predictions from expressions of individual genes; it is usually difficult to find relationships among the genes involved in such a classifier and to interpret the latter in terms of biological processes.

Lifting features to the set level incurs a significant compression of the training data since the number of considered gene sets is typically much smaller than the number of interrogated genes. This compression raises the natural question whether relevant information is lost in the transformation, and whether the augmented interpretability will be outweighed by compromised predictive accuracy. On the other hand, reducing the number of sample features may mitigate the risk of overfitting and thus, conversely, contribute to higher accuracy. In machine learning terms, reformulation of data samples through set-level features increases the *bias* and decreases the *variance* of the learning process [8]. An objective of this study is to assess experimentally the combined effect of the two antagonistic factors on the resulting predictive accuracy.

Another aspect of transforming features to the set level is that biological background knowledge is channeled into learning through the prior definitions of biologically plausible gene sets. Among the goals of this study is to assess how significantly such background knowledge contributes to the performance of learned classifiers. We do this assessment by comparing classification accuracy achieved with genuine curated gene sets against that obtained with gene sets identical to the latter in number and sizes, yet lacking any biological relevance. We also investigate patterns distinguishing genuine gene sets particularly useful for classification from those less useful.

A further objective is to evaluate–from the machine learning perspective–statistical techniques proposed recently in the research on set-level gene expression analysis. These are the Gene Set Enrichment Analysis (GSEA) method [1], the SAM-GS algorithm [3] and a technique known as the Global test [2]. Informally, they rank a given collection of gene sets according to their correlation with phenotype classes. The methods naturally translate into the machine learning context in that they facilitate feature selection [23], i.e. they are used to determine which gene sets should be provided as sample features to the learning algorithm. We experimentally verify whether these methods work reasonably in the classification setting, i.e. whether learning algorithms produce better classifiers from gene sets ranked high by the mentioned methods than from those ranking lower. We investigate classification conducted with a single selected gene set as well as with a batch of high ranking sets. Furthermore, we test how the three gene-set-specific methods compare to some generic feature selection heuristics (information gain and support vector machine with recursive feature extraction) known from machine learning.

To use a machine learning algorithm, a unique value for each feature of each training sample must be established. Set-level features correspond to multiple expressions and these must therefore be aggregated. We comparatively evaluate three aggregation options. The first (AVG) simply averages the expressions of the involved genes. The value assigned to a sample and a gene set is independent of other samples and classes. The other two, more sophisticated, methods (SVD, SetSig) rely respectively on the singular value decomposition principle [7] and the so-called gene set signatures [22]. In the latter two approaches, the value assigned to a given

sample and a gene set depends also on expressions measured in other samples. Let us return to the initial experimental question concerned with how the final predictive accuracy is influenced by the training data compression incurred by reformulating features to the set level. As follows from the above, two factors contribute to this compression: selection (not every gene from the original sample representation is a member of a gene set used in the set-level representation, i.e. some interrogated genes become ignored) and aggregation (for every gene set in the set-level representation, expressions of all its members are aggregated into a single value). We quantify the effects of these factors on predictive accuracy. Regarding selection, we experiment with set-level representations based on 10 best gene sets and 1 best gene set, respectively, with both numbers chosen ad-hoc. The two options are applied with all three selection methods (GSEA, SAM-GS, Global). We compare the obtained accuracy to the baseline case where all individual genes are provided as features to the learning algorithm, and to an augmented baseline case where a prior feature-selection step is taken using the information gain heuristic. For each of the selection cases, we further evaluate the contribution of the aggregation factor. This evaluation is done by comparing all the three aggregation mechanisms (AVG, SVD, SetSig) to the control case where no aggregation is performed at all; in this case, individual genes combined from the selected gene groups act as features.

The key contribution of the present study is thus a thorough experimental evaluation of a number of aspects and methods of the set-level strategy employed in the machine learning context, entailing the reformulation of various, independently published relevant techniques into a unified framework. Such a contribution is important both due to the current state of the art in microarray data analysis, wherein according to the review [24], *the need for thoroughly evaluating existing techniques currently seems to outweigh the need to develop new techniques*, and specifically due to the inconclusive results of previous, less extensive studies indicating both superiority (e.g. [20]) and inferiority (Section 4 in [22]) of the set-level approach to classificatory machine learning, with respect to the accuracy achievable by the baseline gene-level approach.

Our contributions are, however, also significant beyond the machine learning scope. In the general area of set-level expression analysis, it is undoubtedly important to establish a performance ranking of the various statistical techniques for the identification of significant gene sets in class-labeled expression data. This is made difficult by the lack of an unquestionable ranking criterion–there is in general no ground truth stipulating which gene sets should indeed be identified by the tested algorithms. The typical approach embraced by comparative studies such as [3] is thus to appeal to intuition (e.g. *the p53 pathway should be identified in p53-gene mutation data*). However legitimate such arguments are, evaluations based on them are obviously limited in generality and objectivity. We propose that the predictive classification setting supported by the cross-validation procedure for unbiased accuracy estimation, as adopted in this paper, represents exactly such a needed framework enabling objective comparative assessment of gene set selection techniques. In this framework, results of gene set selection are deemed good if the selected gene sets allow accurate classification of new samples. Through cross-validation, the accuracy can be estimated in an unbiased manner.

Main results

We first verified whether gene sets ranked high by the established set-level analysis methods (GSEA, SAM-GS, Global) indeed lead to construction of better classifiers by machine learning algorithms, i.e. we investigated how classification accuracy depends on Factor 3 in Table 1. In the top panel of Figure 1, we plot the average accuracy for Factor 3 alternatives ranging 1 to 10 (top 10 gene sets), and n – 9 to n (bottom 10). The trend line fitted by the least squares method shows a clear decay of accuracy as lower-ranking sets are used for learning. The bottom panel corresponds to Factor 3 values 1:10 (left) and n – 9 : n (right) corresponding to the situations where the 10 top-ranking and the 10 bottom-ranking (respectively) gene sets are combined to produce a feature set for learning. Again, the dominance of the former in terms of accuracy is obvious.

Given the above, there is no apparent reason why low-ranking gene sets should be used in practical experiments. Therefore, to maintain relevance of the subsequent conclusions, we conducted further analyses on

Table 1 Factors

Analyzed factors	***Alternatives***	***#Alts***
1. Gene sets (Sec.)	Genuine, Random	2
2. Ranking algo (Sec.)	GSEA, SAM-GS, Global	3
*3. Set(s) forming features**	1, 2, ... 10,n - 9, n - 8,...n,1:10, n - 9 : n	22
4. Aggregation (Sec.)	SVD, AVG, SetSig, None	4
Product		528
Auxiliary factors	***Alternatives***	***#Alts***
5. Learning algo (Sec.)	svm, 1-nn, 3-nn, nb, dt	5
6. Dataset (Sec.)	d_1 ... d_{30}	30
7. Testing Fold	f_1 ... f_{10}	10
Product		1500

Alternatives considered for factors influencing the set-level learning workflow. The number left of each factor refers to the workflow step (Fig. 2) in which it acts.

*Identified by rank, n corresponds to the lowest ranking set, i:j denotes that all of gene sets ranking i to j are used to form features.

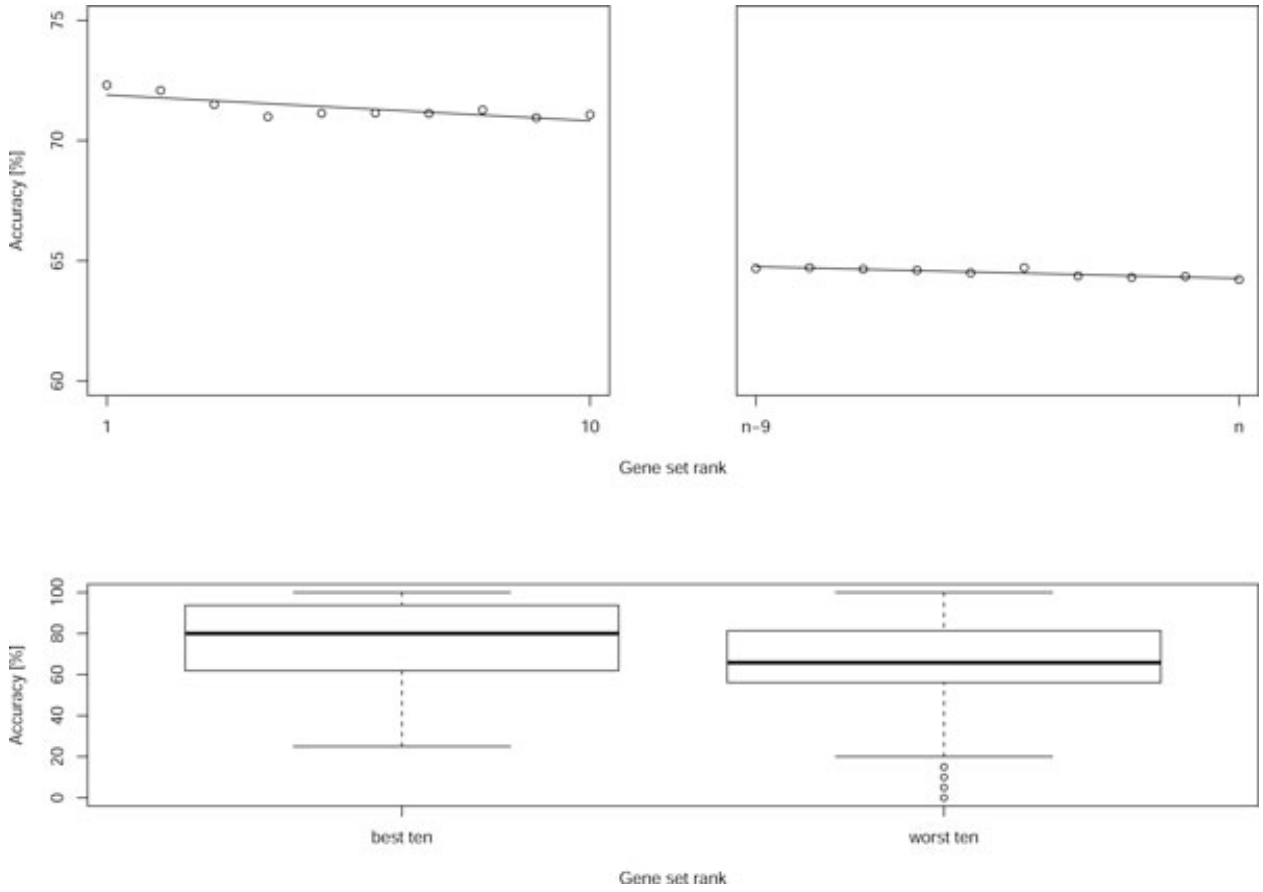


Figure 1 Accuracy decay. The top panels show the plots for the average accuracy of Factor 3 alternatives ranging 1 to 10, and n−9 to n. Average predictive accuracy tends to fall as lower-ranking gene sets are used to constitute features (see text for details). The trend lines shown in the top panels are the ones minimizing the residual least squares. The bottom panel gives the accuracy boxplot for the batch experiments. 10 highest-ranking and the 10 lowest-ranking (respectively) gene sets are combined to produce a feature set for learning. Again, the dominance of the former in terms of accuracy is obvious. Each point in the top panels and each box plot in the bottom panel follows from 16,000 learning experiments.

the set-level experimental sample only with measurements where Factor 3 (gene set rank) is either 1 or 1:10.

We next addressed the hypothesis that genuine gene sets constitute better features than random gene sets, i.e. we investigated the influence of Factor 1 in Table 1. Classifiers learned with genuine gene sets exhibited significantly higher predictive accuracies ($p = 1.4 \times 10^{-4}$, one-sided test) than those based on random gene sets.

Given this result, there is a clear preference to use genuine gene sets over random gene sets in practical applications. Once again, to maintain relevance of our subsequent conclusions, we constrained further analyses of the set-level sample to measurements conducted with genuine gene sets.

Working now with classifiers learned with high-ranking genuine gene sets, we revisited Factor 3 to assess the difference between the remaining alternatives 1 and 1:10 corresponding respectively to more and less compression of training data. The 1:10 variant where sample features capture information from the ten best gene sets exhibits significantly ($p = 3.5 \times 10^{-5}$) higher accuracy than the 1 variant using only the single best gene set to constitute features (that is, a single feature if aggregation is employed).

We further compared the three dedicated gene-set ranking methods, i.e. evaluated the effect of Factor 2 in Table 1. Since three comparisons are conducted in this case (one per pair), we used the Bonferroni-Dunn adjustment on the Wilcoxon test result. The Global test turned out to exhibit significantly higher accuracy than either SAM-GS ($p = 0.0051$) or GSEA ($p = 0.0039$). The difference between the latter two methods was not significant.

Concerning the aggregation method (Factor 4 in Table 1), there are two questions of interest: whether there are significant differences in the performance of

the individual aggregation methods (SVD, AVG, SetSig), and whether aggregation in general has a detrimental effect on performance. As for the first question, both SVD and SetSig proved to outperform AVG ($p = 0.011$ and $p = 0.03$, respectively), while the difference between SVD and SetSig is insignificant. The answer to the second question turned out to depend on Factor 3 as follows. In the more compressive (1) alternative, the answer is affirmative in that all the three aggregation methods result in less accurate classifiers than those not involving aggregation ($p = 0.0061$ for SVD, $p = 0.013$ for SetSig and $p = 1.1 \times 10^{-4}$ for AVG, all after Bonferroni-Dunn adjustment).

However, the detrimental effect of aggregation tends to vanish in the less compressive (1:10) alternative of Factor 3, where only the AVG alternative in comparison to None yields a significant difference ($p = 0.011$). Table 2 summarizes the main findings presented above.

The principal trends can also be well observed through the ranked list of methodological combinations by median classification accuracy, again generated from measurements not involving random or low-ranking gene sets. This is shown in Table 3. Position 17 refers to the baseline method where sample features capture expressions of all genes and prior gene set definitions are ignored. In agreement with the statistical conclusions above, the ranked table clearly indicates the superiority of the Global test for gene-set ranking, and of using the 10 best gene sets (i.e., the 1:10 alternative) to establish features rather than relying only on the single best gene set. It is noteworthy that all four combinations involving the Global test and the 1:10 alternative (i.e., ranks 1, 2, 4, 5) outperform the baseline method.

While intuitive, rankings based on median accuracy over multiple datasets may, according to [25], be problematic as to their statistical reliability. Therefore, we offer in Table 4 an alternative ranking of the 19 methods that avoids mixtures of predictive accuracies from different datasets. Here, the methods were sub-ranked on each of the 150 combinations of 30 datasets and 5 learning algorithms by cross-validated predictive accuracy achieved on that combination. The 150 sub-ranks were then averaged for each method, and this average dictates the ranking shown in the table. In this ranking, the baseline strategy improves its rank to Position 5. The superiority of classifiers learned from 10 gene sets selected by the Global test, as formerly noted for Table 3, continues to hold in the alternative ranking underlying Table 4.

Table 2 Summary of results

Factor	*Alternatives*	
	Better	***Worse***
1. Gene sets	Genuine	Random
2. Ranking algo	Global	SAM-GS, GSEA
3. Sets forming features	high ranking, 1:10 (best ten sets)	low ranking, 1 (best set)
*4. Aggregation**	**SetSig**, **SVD**	**AVG**

See Section *Main Results* for details on how the conclusions were determined.
*Difference not significant if Factor 3 is 1:10.

Table 3 Ranking of gene set methods

Rank	*Methods*			*Accuracy*			
	Sets	***Rank. Algo***	***Aggrgt***	***Median***	***Avg***	***σ***	***Iqr***
1	1:10	Global	SVD	89.2	79.5	18.9	33.2
2	1:10	Global	None	88.3	81.0	17.7	31.3
3	1	Global	None	87.8	80.7	17.5	31.0
4	1:10	Global	SetSig	87.4	81.1	16.5	26.1
5	1:10	Global	AVG	85.6	78.7	18.4	32.6
6	1:10	SAM-GS	SetSig	85.4	79.9	17.1	30.2
7	1:10	SAM-GS	None	84.6	80.1	17.3	30.7
8	1	Global	SVD	83.8	77.9	20.1	34.3
9	1:10	GSEA	SetSig	83.4	78.3	16.7	26.3
10	1:10	GSEA	None	82.3	80.0	16.8	30.4
11	1:10	SAM-GS	SVD	79.9	77.1	18.0	32.1
12	1:10	GSEA	SVD	79.2	77.2	17.7	31.7
13	1:10	GSEA	AVG	79.1	76.4	16.9	31.9
14	1	SAM-GS	None	78.3	76.0	15.3	26.3
15	1	Global	SetSig	77.5	75.9	15.1	23.5
16	1	GSEA	None	76.7	75.6	16.3	29.5
17	*baseline (all genes used)*			75.5	76.6	18.4	33.5
18	1	SAM-GS	SetSig	75.0	74.7	14.2	18.9
19	1	Global	AVG	72.7	73.8	17.6	31.1
20	1:10	SAM-GS	AVG	72.5	73.8	15.9	26.0
21	1	GSEA	SetSig	70.2	72.6	17.0	26.8
22	1	GSEA	AVG	69.6	68.1	12.8	22.4
23	1	GSEA	SVD	69.5	71.9	16.3	28.2
24	1	SAM-GS	SVD	69.0	69.5	15.7	21.3
25	1	SAM-GS	AVG	67.3	67.0	11.4	15.5

Ranking of combinations of gene set methods by median predictive accuracy achieved on 30 datasets (Table 8, Section *Expression and gene sets*) with 5 machine learning algorithms (Section *Machine learning*) estimated through 10-fold cross-validation (i.e. 1,500 experiments per row). The columns indicate, respectively, the resulting rank by median accuracy, the gene sets used to form features (1 - the top ranking set, 1:10 - the top ten ranking sets), the gene set selection method, the expression aggregation method (see Section *Methods and data* for details on the latter 3 factors), and the median, average, standard deviation and interquartile range of the accuracy.

Additional analyses

Generic feature selection

In the set-level classification framework, gene sets play the role of sample features. Therefore the three gene-set ranking methods (GSEA, SAM-GS, Global) are employed for feature selection conducted in the learning workflow. While the latter three methods originate from research on gene expression analysis, generic feature selection methods have also been proposed in machine learning research [23]. It is interesting to compare the latter to the gene-expression-specific methods. To this end, we

Table 4 Ranking of all combinations of methods

Rank	Methods			Avg Subrank
	Sets	Rank. algo	Aggrgt	
1	1:10	Global	None	15.3
2	1:10	Global	SetSig	15.7
3	1	Global	None	16.3
4	1:10	GSEA	None	16.7
5	*baseline (all genes used)*			16.8
6	1:10	Global	SVD	17.0
7	1:10	SAM-GS	None	17.2
8	1:10	SAM-GS	SetSig	17.6
9	1:10	Global	AVG	18.6
10	1	Global	SVD	19.4
11	1:10	GSEA	SetSig	19.9
12	1:10	GSEA	SVD	20.1
13	1:10	SAM-GS	SVD	20.8
14	1:10	GSEA	AVG	22.1
15	1	Global	SetSig	22.2
16	1	SAM-GS	None	23.0
17	1	SAM-GS	SetSig	23.8
18	1	GSEA	None	23.9
19	1	Global	AVG	24.6
20	1:10	SAM-GS	AVG	25.5
21	1	GSEA	SVD	26.7
22	1	GSEA	SetSig	26.8
23	1	SAM-GS	SVD	28.3
24	1	SAM-GS	AVG	30.3
25	1	GSEA	AVG	30.9

Ranking of all combinations of methods in terms of average subrank. Subranking is done on each of the 150 combinations of 30 datasets and 5 learning algorithms by cross-validated predictive accuracy. Column descriptions are as in Table 3.

consider two approaches. *Information Gain* (IG) [10] is a feature-selection heuristic popular in machine learning. In brief, IG measures the expected reduction in class-entropy caused by partitioning the given sample set by the values of the assessed feature. One of the main disadvantages of IG is that it disregards potential feature interactions. *Support Vector Machine with Recursive Feature Extraction* (SVM-RFE) [26] is a method that ranks features by repetitive training of a SVM classifier with a linear kernel while gradually removing the feature with the smallest input classifier weight. This approach does not assume that features are mutually independent. On the other hand, it naturally tends to select a feature set that maximizes the accuracy of the specific kind of classifier (SVM). For computational reasons (large number of runs and genes), we removed several features at a time ($F \times 2^{-i}$ features in the i-th iteration, where F is the original number of features). [26] mentions such a modification with the caveat that it may be at the expense of possible classification performance degradation.

In the present context, generic feature selection can be applied either on the gene level or on the set level. We explored both scenarios.

The gene-level application produces a variant of the baseline classifier (position 17 in Table 3, position 5 in Table 4) where, however, the learning algorithm only receives features corresponding to genes top-ranked by the feature selection heuristic, rather than all measured genes. The selection is thus based only on the predictive power of the individual genes and ignores any prior definitions of gene sets. The question of how many top-ranking genes should be used for learning is addressed as follows. We want to make the resulting predictive accuracy comparable to that obtained in the main (set-level) experimental protocol, in particular to the 1 and 1:10 alternatives of Factor 3. The median of the number of unique genes present in the selected gene sets in the 1 (1:10, respectively) alternative is 22 (228). Therefore we experiment respectively with 22 and 228 genes top-ranked by generic feature selection. The results are shown in Table 5. Comparing the latter to Tables 3 and 4, we observe that both variants improve the baseline and in fact produce the most accurate classifiers (IG outperforms the set-level approaches, SVM-RFE is comparable with the Global test). SVM-RFE does not outperform IG in general, but it does so in the special case when SVM is used as the learning algorithm.

While the gene-level application of feature selection results in accurate classifiers, the obvious drawback of this approach is that the genes referred in such produced classifiers cannot be jointly characterized by a biological concept. This deficiency is removed if feature selection is instead applied on the set level, i.e. to rank apriori-defined gene sets. This way, the selection methods essentially become the fourth and fifth alternative of Factor 2 (see Table 1) up to the following nuance. While the dedicated gene-set methods (GSEA, SAM-GS, Global) score a feature (gene set) by the expressions of its multiple member genes, IG and SVM-RFE score a feature by the single real value assigned to it, i.e., by the aggregated expressions of the member genes. Therefore,

Table 5 Generic feature selection (gene-level)

# Method	# Selected Genes	Accuracy				Avg Subrank
		Median	Avg	σ	Iqr	
IG	22	90.2	81.5	18.1	30.7	15.0
IG	228	89.8	82.0	17.9	30.3	14.5
SVM-RFE	228	88.3	82.3	16.7	28.5	16.4
SVM-RFE	22	88.0	82.1	17.2	30.4	16.2

Performance of the baseline classification method equipped with a feature-selection step prior to learning. Features (genes) are ranked by the information gain and SVM-RFE heuristics. The number of selected top-ranking genes (22 and 228, respectively) corresponds to the mean number of unique genes acting in gene sets selected in the 1 and 1:10 (respectively) alternatives of the set-level workflow.

when using the generic feature selection, the aggregation step in the experimental workflow (Figure 2) must precede the ranking step. The results of applying IG and SVM-RFE on the set level are shown in Table 6. Comparing again to Tables 3 and 4, both IG and SVM-RFE are outperformed by the Global test (Wilcoxon test, p = 0.017).

Table 6 Generic feature selection (set-level)

Sets	*Methods*		*Accuracy*				*Avg Subrank*
	Selection	*Aggrgt*	*Median*	*Avg*	*σ*	*Iqr*	
1:10	SVM-RFE	SVD	88.3	80.6	17.3	33.0	17.6
1:10	IG	SVD	87.0	79.0	18.7	31.6	17.4
1:10	IG	AVG	84.6	78.2	18.6	33.4	18.7
1:10	SVM-RFE	AVG	84.4	79.2	17.1	31.2	19.2
1:10	SVM-RFE	SetSig	82.5	78.7	17.0	31.2	19.4
1	IG	SVD	80.8	76.3	17.7	33.1	22.5
1:10	IG	SetSig	80.0	77.1	17.4	33.2	20.8
1	SVM-RFE	SetSig	71.8	73.7	15.8	26.4	23.3
1	SVM-RFE	SVD	71.5	74.4	17.4	30.3	23.0
1	IG	AVG	70.9	74.0	18.6	33.1	24.1
1	SVM-RFE	AVG	70.8	72.5	15.4	26.6	24.4
1	IG	SetSig	66.2	68.8	16.2	25.0	28.9

Performance of the set level classification strategy using the information gain and SVM-RFE heuristics for ranking gene sets. Column descriptions are as in Table 3.

Successful gene sets

We also explored patterns distinguishing gene sets particularly useful for classification from other employed gene sets sourced from the Molecular Signatures Database. To this end, we defined three groups of gene sets. The first group referred to as *full* comprises the entire collection of 3028 gene sets obtained from the database (gene sets containing fewer than 5 or more than 200 genes were discarded). The second group referred to as *selected* consists of the 900 gene sets ranked high (1^{st} to 10^{th}) by any of the three selection methods for any of the dataset. The third group referred to as *successful* is a subset of the *selected* group and contains the 210 gene sets acting in classifiers that outperformed the baseline.

We investigated two kinds of properties of the gene sets contained in the three respective groups. First, we considered the gene set type as defined in the Molecular

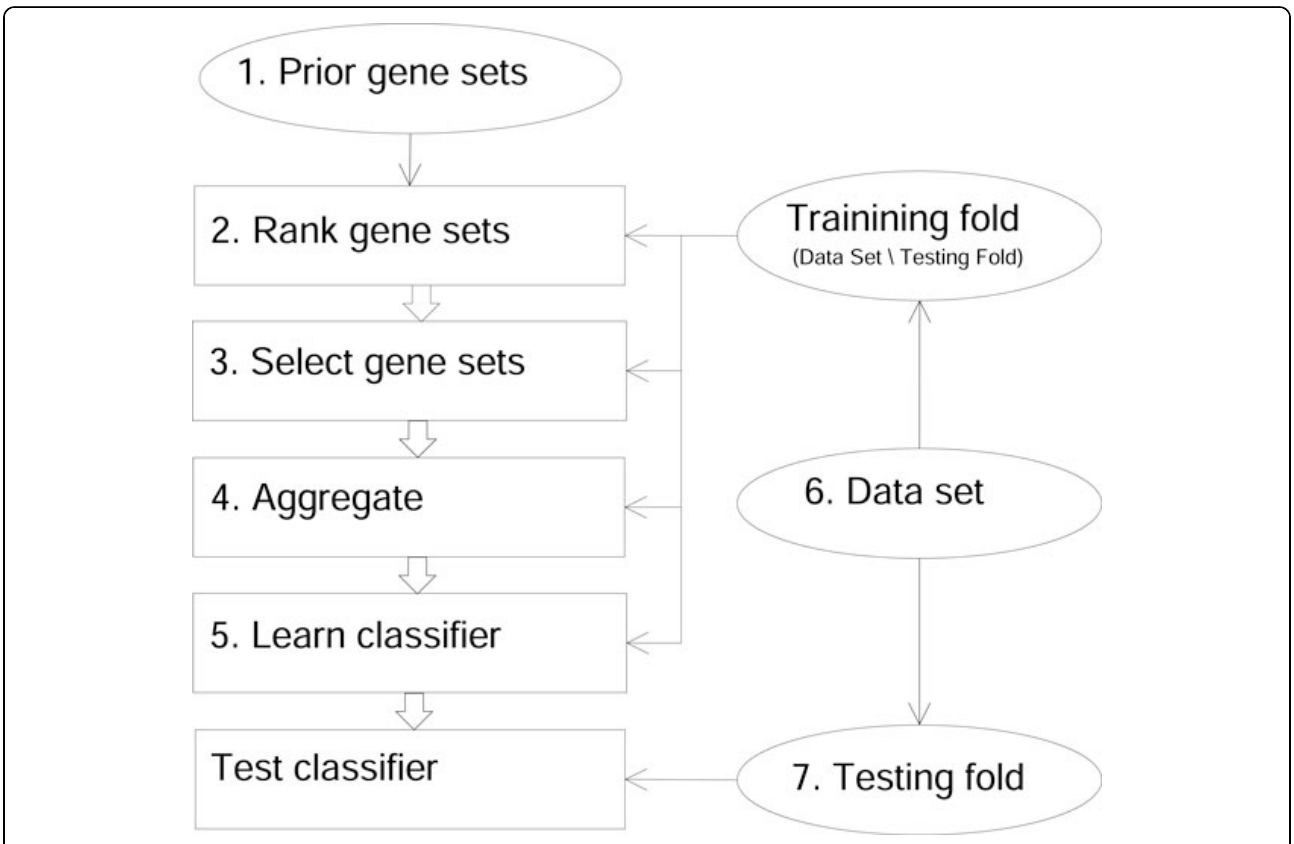


Figure 2 Workflow. The workflow of a set-level learning experiment conducted multiple times with varying alternatives in the numbered steps. For compatibility with the learned classifier, testing fold samples are also reformulated to the set level. The reformulation is done using gene sets selected in Step 3 and aggregation algorithm used in Step 4. The diagram abstracts from this operation.

Signatures Database. The gene sets belonging to the category of chemical and genetic perturbations (CGP) were more frequently *selected* and also more frequently appeared in the *successful* group than the gene sets representing canonical pathways (CP) (full: CGPs 73%, CPs 27%, selected: CGPs 88%, CPs 12%, successful: CGPs 88%, CPs 12%). Second, we considered four possible notions of gene set *size*: i) nominal size (the gene set cardinality), ii) effective size (number of genes from the gene set measured in the dataset), iii) number of PCA coefficients capturing 50% of expression variance in the gene set, iv) as in iii) but with 90% variance. As follows from Table 7, the *successful* group contains smaller gene sets than the other two groups, and this trend is most pronounced for the Global test ranking method (Mann-Whitney U test, the *successful* group versus the *full* group, Bonferroni adjustment: Effective size $p = 0.084$, PCA 90% $p = 0.0039$).

Conclusions and discussion

Set-level approaches to gene expression data analysis have proliferated in the last years, evidence of which are both theoretical studies [1,2] and software tools with set-level functionalities [27] such as enrichment analysis. The added insight and augmented interpretability of analysis results are the main reasons for the popularity of the set-level framework. For the same reasons, the framework has recently been also explored in the context of predictive classification of gene expression data through machine learning [4,17-22]. Conclusions of such studies have however been rather limited as to the range of classification problems considered and techniques used in the set-level machine learning workflow, and inconclusive as to the statistical performance of set-level classifiers. To this end, we have presented a large experimental study, in which we formalized the mentioned set-level workflow, identified various independently published techniques relevant to its individual steps, and reformulated them into a unified framework. By executing various instantiations of the workflow on 30 gene expression classification problems, we have established the following main conclusions.

1. State-of-the-art gene set ranking methods (GSEA, SAM-GS, Global test) perform sanely as feature selectors in the machine learning context in that high ranking gene sets outperform (i.e., constitute better features for classification than) those low ranking.
2. Genuine curated gene sets from the Molecular Signature Database outperform randomized gene sets. Smaller gene sets and sets pertaining to chemical and genetic perturbations were particularly successful.
3. For gene set selection, the Global test [2] outperforms each of SAM-GS [3], GSEA [1] as well as the generic information gain heuristic [10] and the SVM-based recursive feature elimination approach [26].
4. For aggregating expressions of set member genes into a unique feature value, both SVD [7] and SetSig [22] outperform arithmetic averaging [4].
5. Using top ten gene sets to construct features results in better classifiers than using only the single best gene set.
6. The set-level approach using top ten genuine gene sets as ranked by the Global test outperforms the baseline gene-level method in which the learning algorithm is given access to expressions of all measured genes. However, it is outperformed by the baseline approach if the latter is equipped with a prior feature selection step.

Conclusion 1 is rather obvious and was essentially meant as a prior sanity check.

Table 7 Comparison of the full, selected and successful group of gene sets

Group	*Selection*	*Statistic*	*Nominal size*	*Effective size*	*PCA 50% var*	*PCA 90% var*
Full	None	mean	71.7±1.7	40.9±0.7	4.4±0.03	16.7±0.14
		median	37.0	28.1	4.1	15.3
Selected	all	mean	62.5±2.7	47.8±1.9	3.8±0.08	15.1±0.35
		median	33.5	27.0	3.4	13.4
	Global	median	32.0	25.5	3.3	12.8
	GSEA	median	34.0	27.0	3.4	13.7
	SAM-GS	median	40.5	28.0	3.7	14.3
Successful	all	mean	56.9±4.4	39.2±2.9	4.3±0.14	14.7±0.56
		median	31.0	21.0	3.9	12.6
	Global	median	22.0	18.5	3.8	11.7
	GSEA	median	37.0	27.5	4.3	14.2
	SAM-GS	median	30.5	22.5	4.0	12.7

Mean and median sizes of gene sets partitioned into three groups (see Section *Successful gene sets* for details.)

The first statement of Conclusion 2 is not obvious, since constructing randomized gene sets in fact corresponds to the machine learning technique of stochastic feature extraction [28] and as such may itself contribute to learning good classifiers. Nevertheless, relevant background knowledge resting in the prior definition of biologically plausible gene sets contributes further to increasing the predictive accuracy. Conclusions 3 and 4 are probably the most significant for practitioners in set-level predictive modeling of gene expression as so far there has been no clear guidance to choose from the two triples of methods.

Concerning Conclusion 3, the advantages of the Global test were argued in [2] but not supported in terms of the predictive power of the selected gene sets. As for conclusion 4, the SetSig technique was introduced and tested in [22], appearing superior to both averaging and a PCA-based method which is conceptually similar to the SVD method [7]. However, owing to the limited experimental material in [22], the ranking was not confirmed by a statistical test. Here we confirmed the superiority of SetSig with respect to averaging, however, the difference of in the performance of SetSig and SVD was not significant.

A further remark concerns the mentioned aggregation methods. All three of them are applicable to any kind of gene sets, whether these are derived from pathways, gene ontology or other sources of background knowledge. The downside of this generality is that substantial information available for specific kinds of gene sets is ignored. Of relevance to pathway-based gene sets, the recent study [29] convincingly argues that the perturbation of a pathway depends on the expressions of its member genes in a non-uniform manner. It also proposes how to quantify the impact of each member gene on the perturbation, given the graphical structure of the pathway. It seems reasonable that a pathway-specific aggregation method should also weigh member genes by their estimated impact on the pathway. Such a method would likely result in more informative pathway-level features and could outperform the three aggregation methods we have considered.

Conclusion 5 is not entirely surprising. Relying only on a single gene set entails too large an information loss and results in classifiers less accurate than those using ten best gene sets. Note that in the single gene set case, when aggregation is applied (i.e., Factor 4 in Table 1 is other than None, see the first example in Figure 3), the sample becomes represented by only a single real-valued feature and learning essentially reduces to finding a threshold value for it. To verify that more than one gene set should be taken into account, we tested the 10-best-sets option and indeed it performed better. Obviously, the optimal number of sets to be considered depends on the particular classification problem and data, and in practice it can be estimated empirically, e.g. through internal cross-validation. Here, training data T would be randomly split into a validation set V and the remainder $T' = T \setminus V$, e.g. with the 20%-80% proportion. Classifiers would first be learned with T', each with a different value for the number of gene sets forming features; this number could range e.g. as $f \in \{2, 4, 8,..., 128\}$. The number f^* yielding the classifier most accurate on the validation set V is then an estimate of the optimal number of features. The final classifier would then be learned on the entire training set T, using f^* features. While we could not follow this procedure due to computational considerations (the already high number of learning sessions would have grown excessively), it is a reasonable instrument in less extensive experiments such as in single-domain classification.

A straightforward interpretation of Conclusion 6 is that the set-level framework is not an instrument for boosting predictive accuracy. However, set-level classifiers have a value per se, just as set-level units are useful in standard differential analysis of gene expression data. In this light, it is important that with a suitable choice of techniques, set-level classifiers do achieve accuracy competitive with conventional gene-level classifiers.

Methods and data

Here we first describe the methods adopted for gene set ranking, gene expression aggregation, and for classifier learning. Next we present the datasets used as benchmarks in the comparative experiments. Lastly, we describe the protocol followed by our experiments.

Gene set ranking

Three methods are considered for ranking gene sets. As inputs, all of the methods take a set $G = \{g_1, g_2,...g_p\}$ of interrogated genes, and a set S of N expression samples where for each $s_i \in S$, $s_i = (e_{1,i}, e_{2,i},...e_{p,i}) \in \mathbb{R}^p$ where $e_{j,i}$ denotes the (normalized) expression of gene g_j in sample s_i. The sample set S is partitioned into phenotype classes $S = C_1 \cup C_2 \cup ... \cup C_o$ so that $C_i \cap C_j = \{\}$ for $i \neq j$. To simplify this paper, we assume binary classification, i.e. $o = 2$. A further input is a collection of gene sets $\mathcal{G}$ such that for each $\Gamma \in \mathcal{G}$ it holds $\Gamma \subseteq G$. In the output, each of the methods ranks all gene sets in $\mathcal{G}$ by their estimated power to discriminate samples into the predefined classes.

Next we give a brief account of the three methods and refer to the original sources for a more detailed description. In experiments, we used the original implementations of the procedures as provided or published by the respective authors.

Gene Set Enrichment Analysis (GSEA) [1] tests a null hypothesis that gene rankings in a gene set Γ,

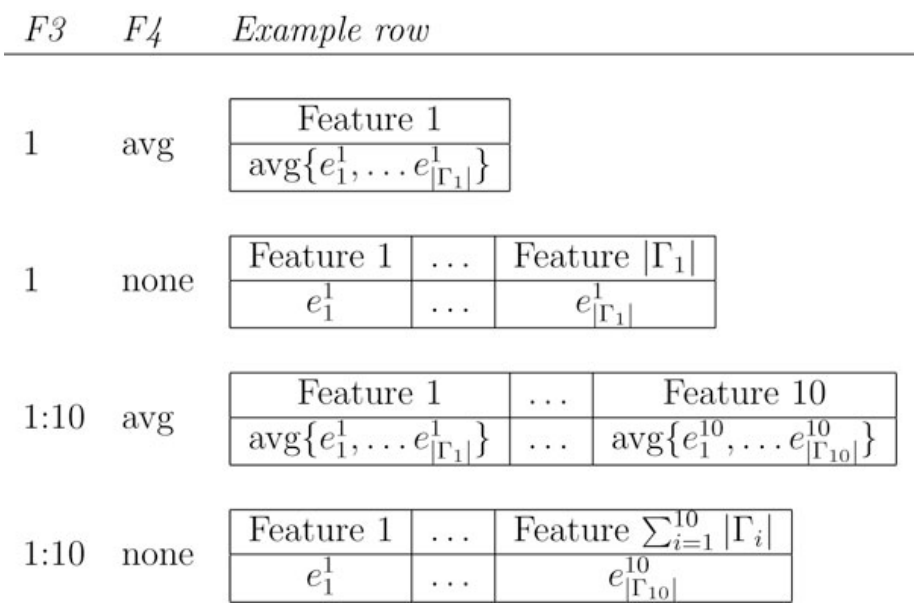


Figure 3 Examples of sample representation. Examples of sample representation generated with four combinations of alternatives of factors 3 and 4 from Table 1. Shown for one sample (i.e. header + one row) with e^j_i denoting the expression of the *i*-th member of the *j*-ranked gene set Γ_j. Non-exemplified combinations of the two factors are analogical to the cases shown. The remaining considered factors do not influence the structure of sample representation.

according to an association measure with the phenotype, are randomly distributed over the rankings of all genes. It first sorts G by correlation with binary phenotype. Then it calculates an enrichment score (ES) for each $\Gamma \in \mathcal{G}$ by walking down the sorted gene list, increasing a running-sum statistic when encountering a gene $g_i \in \Gamma$ and decreasing it otherwise. The magnitude of the change depends on the correlation of g_i with the phenotype. The enrichment score is the maximum deviation from zero encountered in the random walk. It corresponds to a weighted Kolmogorov-Smirnov-like statistic. The statistical significance of the ES is estimated by an empirical phenotype-based permutation test procedure that preserves the correlation structure of the gene expression data. GSEA was one of the first specialized gene-set analysis techniques. It has been reported to attribute statistical significance to gene sets that have no gene associated with the phenotype, and to have less power than other recent test statistics [2,3].

SAM-GS [3]

This method tests a null hypothesis that the mean vectors of the expressions of genes in a gene set do not differ by phenotype. Each sample s_i is viewed as a point in an N -dimensional Euclidean space. Each gene set $\Gamma \in \mathcal{G}$ defines its $|\Gamma|$-dimensional subspace in which projections s_i^Γ of samples s_i are given by coordinates corresponding to genes in Γ. The method judges a given by how distinctly the clusters of points $\{s_i^\Gamma | s_i \in C_1\}$ and $\{s_j^\Gamma | s_j \in C_2\}$ are separated from each other in the subspace induced by Γ. SAM-GS measures the Euclidean distance between the centroids of the respective clusters and applies a permutation test to determine whether, and how significantly, this distance is larger than that obtained if samples were assigned to classes randomly.

The Global Test [2]

The global test, analogically to SAM-GS, projects the expression samples into subspaces defined by gene sets $\Gamma \in \mathcal{G}$. In contrast to the Euclidean distance applied in **SAM-GS**, it proceeds instead by fitting a regression function in the subspace, such that the function value acts as the class indicator. The degree to which the two clusters are separated then corresponds to the magnitude of the coefficients of the regression function.

Expression aggregation

Three methods are considered for assigning a value to a given gene set Γ for a given sample s_i by aggregation of expressions of genes in Γ.

Averaging (AVG)

The first method simply produces the arithmetic average of the expressions $e_{j,i}$ of all Γ genes $1 \le j \le p$ in sample s_i. The value assigned to the pair (s_i, Γ) is thus independent of samples s_j, $i \neq j$.

Singular Value Decomposition (SVD)

A more sophisticated approach was employed by [7]. Here, the value assigned to (s_i, Γ) depends on

expressions $e_{j,i}$ measured in sample s_i but, unlike in the averaging case, also on expressions $e_{j,k}$ measured in samples s_k, $k \neq i$. In particular, all samples in the sample set S are viewed as points in the $|\Gamma|$-dimensional Euclidean space induced by Γ the same way as explained in Section *Gene set ranking*. Subsequently, the specific vector in the space is identified, along which the sample points exhibit maximum variance. Each point $s_k \in S$ is then projected onto this vector. Finally, the value assigned to (s_i, Γ) is the real-valued position of the projection of s_i on the maximum-variance vector in the space induced by Γ.

Gene Set Signatures (SetSig)

Similarly to the SVD method, the SetSig [22] method assigns to (s_i, Γ) a value depending on expressions both in sample s_i as well as in other samples s_k, $k \neq i$. However, unlike in the previous two aggregation methods, here the value also depends on the class memberships of these samples. In particular, SetSig confines to two-class problems and the value ('signature') assigned to (s_i, Γ) can be viewed as the Student's unpaired t-statistic for the means of two populations of the Pearson correlation coefficients. The first (second) population studies correlation of s_i with the samples from the first (second) class in the space induced by Γ. Intuitively, the signature is positive (negative) if the sample correlates rather with the samples belonging to the first (second) class.

Machine learning

We experimented with five diverse machine learning algorithms to avoid dependence of experimental results on a specific choice of a learning method. These algorithms are explained in depth for example by [8]. In experiments, we used the implementations available in the WEKA software due to [30], using the default settings. None of the methods below is in principle superior to the others, although the first one prevails in predictive modeling of gene expression data and is usually associated with high resistance to noise in data.

Support Vector Machine

Samples are viewed as points in a vector space with coordinates given by the values of its features. A classifier is sought in the form of a hyperplane that separates training samples of distinct classes and maximizes the distance to the points nearest to the hyperplane (i.e. maximizing the *margin*) in that space or in a space of extended dimension into which the original vector space is non-linearly projected.

1-Nearest Neighbor

This algorithm is a simple form of classification proceeding without learning a formal data model. A new sample is always predicted to have the same class as the most similar sample (i.e. the nearest neighbor) available in training data. We use the Euclidean metric to measure the similarity of two samples.

3-Nearest Neighbors

This method is similar to 1-Nearest Neighbor, except that class is determined as one prevailing among the three, rather than one, most similar samples in training data. This method becomes superior to the previous one as noise in data exceeds a certain threshold amount. The threshold value (and thus the optimal number of considered neighbors) is in general not known.

Naive Bayes

A sample is classified into the class that is most probable given the sample's feature values, according to a conditional probability distribution learned from training data on the simplifying assumption that, within each class, all features are mutually independent random variables. Gene expression data usually deviate from this assumption and consequently the method becomes suboptimal.

Decision Tree

A tree-graph model enables to derive a class prediction for a sample by following a path from the root to a leaf of the tree, where the path is determined by outcomes of tests on the values of features specified in the internal nodes of the tree. The tree model is learned from training data and can also be represented as a set of decision rules.

Expression and gene sets

We conducted our experiments using 30 public gene expression datasets, each containing samples categorized into two classes. This collection contains both hard and easy classification problems (see Figure 4). The individual datasets are listed in Table 8 and annotated in more detail in the supplemental material at http://ida.felk.cvut.cz/CESLT.

Besides expression datasets, we utilized a gene set database consisting of 3272 manually curated sets of genes obtained from the Molecular Signatures Database (MSigDB v3.0) [1]. These gene sets have been compiled from various online databases (e.g. KEGG, GenMAPP, BioCarta).

For control experiments, we also prepared another collection of gene sets that is identical to the latter in the number of contained sets and the distribution of their cardinalities. However, the contained sets are assembled from random genes and have no biological significance. The particular method used to obtain the randomized gene sets is as follows. For sampling, we consider the set Σ of all genes occurring in some of the genuine gene sets, formally $\Sigma = \{g | g \in \Gamma, \Gamma \in \mathcal{G}\}$. Then, for each genuine gene set Γ, we sample $|\Gamma|$ genes without replacement uniformly from Σ to constitute the counterpart random gene set Γ'.

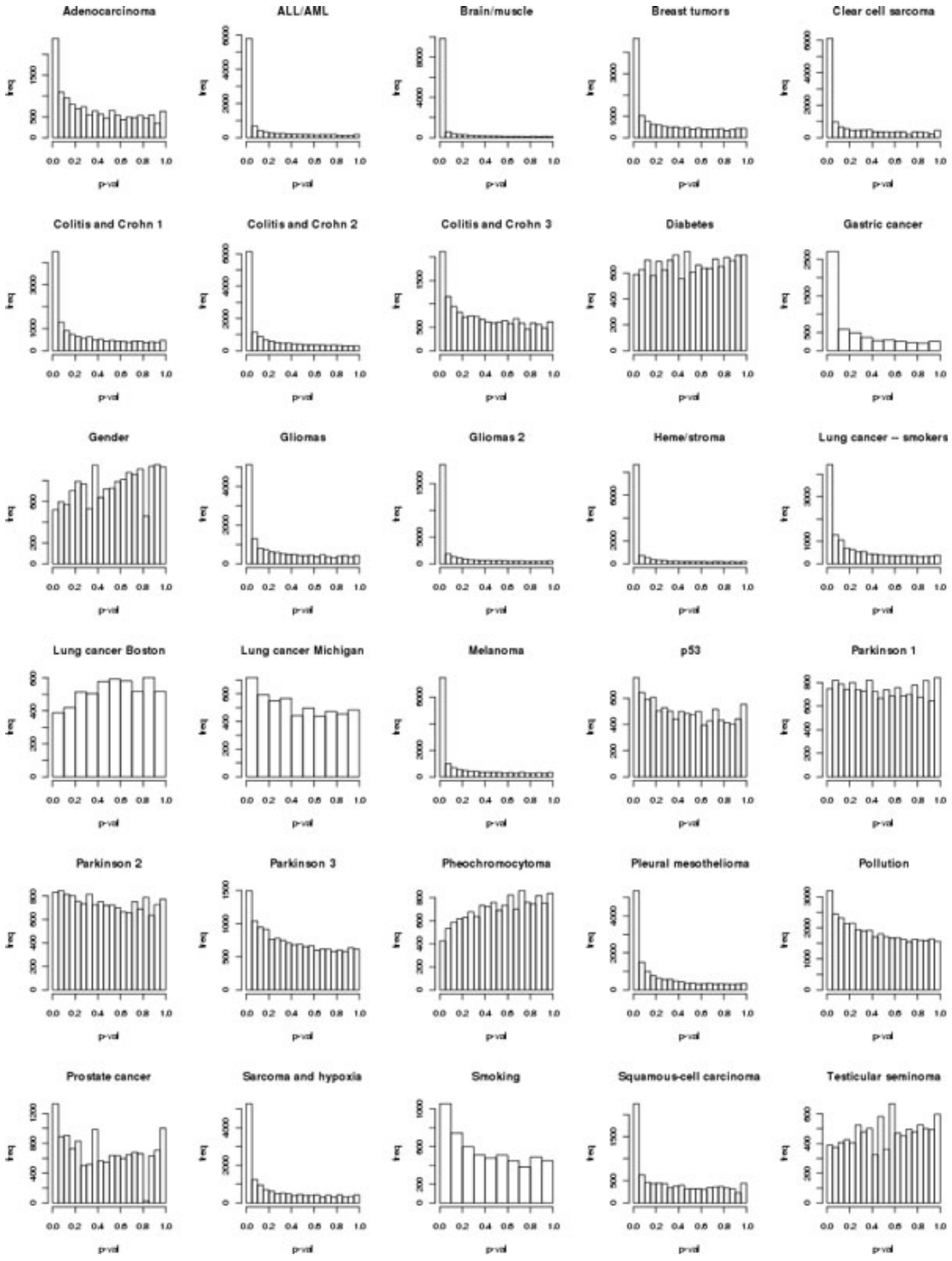


Figure 4 Histograms of differential gene expression. Histograms of differential gene expression suggest the difficulty of the individual domains. An easy domain is supposed to have a strongly left-skewed histogram, while the difficult domains rather show a flat histogram. There is one plot for each of 30 domains, *x* axis shows the p-value of differential expression, the *y* axis gene frequency.

Experimental protocol

Classifier learning in the set-level framework follows a simple workflow. Its performance is influenced by several factors, each corresponding to a particular choice from a class of techniques (such as for gene set ranking). We evaluate the contribution that these factors

Table 8 Datasets

Dataset	*Genes*	*Class 1*	*Class 2*	*Source*	*Reference*
Adenocarcinoma	14023	8	29	GDS2201	[31]
ALL/AML	10056	24	24	Broad institute	[32]
Brain/muscle	13380	41	20	-	[4]
Breast tumors	14023	16	27	GDS1329	[33]
Clear cell sarcoma	14023	18	14	GDS1282	[34]
Colitis and Crohn 1	14902	42	26	GDS1615	[35]
Colitis and Crohn 2	14902	42	59	GDS1615	[35]
Colitis and Crohn 3	14902	26	59	GDS1615	[35]
Diabetes	13380	17	17	Broad institute	[5]
Heme/stroma	13380	18	33	-	[4]
Gastric cancer	5664	8	22	GDS1210	[36]
Gender	15056	15	17	Broad institute	[1]
Gliomas	14902	26	59	GDS1975	[37]
Gliomas 2	31835	23	81	GDS1962	[38]
Lung cancer Boston	5217	31	31	Broad institute	[39]
Lung cancer Michigan	5217	24	62	Broad institute	[40]
Lung cancer - smokers	14023	90	97	GDS2771	[41]
Melanoma	14902	18	45	GDS1375	[42]
p53	10101	33	17	Broad institute	[1]
Parkinson 1	14902	22	33	GDS2519	[43]
Parkinson 2	14902	22	50	GDS2519	[43]
Parkinson 3	14902	33	50	GDS2519	[43]
Pheochromocytoma	14023	38	37	GDS2113	[44]
Pleural mesothelioma	14902	10	44	GDS1220	[45]
Pollution	37804	88	41	-	[46]
Prostate cancer	14023	18	45	GDS1390	[47]
Sarcoma and hypoxia	14902	15	39	GDS1209	[48]
Smoking	5664	18	26	GDS2489	[49]
Squamous-cell carcinoma	9460	22	22	GDS2520	[50]
Testicular seminoma	9460	22	14	GDS2842	[51]

Number of genes interrogated and number of samples in each of the two classes of each dataset.

make to the predictive accuracy of the resulting classifiers by repeated executions of the learning workflow with varying the factors.

The learning workflow is shown in Figure 2. Given a set of binary-labeled training samples from an expression dataset, the workflow starts by ranking the provided collection of a priori-defined gene sets according to their power to discriminate sample classes. The resulting ranked list is subsequently used to select the gene sets which form set-level sample features. Each such feature is then assigned a value for each training sample by aggregating the expressions in the gene set corresponding to the feature. An exception to this pattern is the *None* alternative of the aggregation factor, where expressions are not aggregated, and features correspond to genes instead of gene sets. This alternative is considered for comparative purposes. Figure 3 illustrates the resulting sample representation for four combinations of the selection and aggregation alternatives. Next, a machine learning algorithm produces a classifier from the reformulated training samples. Finally, the classifier's predictive accuracy is calculated as the proportion of samples correctly classified on an independent testing sample fold. For compatibility with the learned classifier, the testing samples are also reformulated to the set level prior to testing, using the same selected gene sets and aggregation mechanism as in the training phase.

Seven factors along the workflow influence its result. The alternatives considered for each of them are summarized in Table 1. We want to assess the contributions of the first four factors (top in table). The remaining three auxiliary factors (bottom in table) are employed to diversify the experimental material and thus increase the robustness of the findings. Factor 7 (testing fold) is involved automatically through the adoption of the 10-fold cross-validation procedure (see e.g. chap. 7 in [8]). We execute the workflow for each possible combination of factor alternatives, obtaining a factored sample of 792,000 predictive accuracy values.

While the measurements provided by the above protocol allow us to compare multiple variants of the set-level framework for predictive classification, we also want to compare these to the baseline gene-level alternative usually adopted in predictive classification of gene expression data. Here, each gene interrogated by a microarray represents a feature. This sample representation is passed directly to the learning algorithm without involving any of the pre-processing factors (1-4 in Table 1). The baseline results are also collected using the 5 different learning algorithms, the 30 benchmark datasets and the 10-fold cross-validation procedure (i.e. Factors 5-7 in Table 1 are employed). As a result, an additional sample of 1,500 predictive accuracy values is collected for the baseline variant.

Finally, to comply with the standard application of the cross-validation procedure, we averaged the accuracy values corresponding to the 10 cross-validation folds for each combination of the remaining factors. The subsequent statistical analysis thus deals with a sample of 79,200 and 150 measurements for the set-level and baseline experiments, respectively, described by the predictive accuracy value and the values of the relevant factors.

All statistical tests conducted were based on the paired Wilcoxon test (two-sided unless stated otherwise). For pairing, we always related two measurements equal in terms of all factors except for the one investigated. The stronger t-test is more usual in analysis of

predictive accuracy samples in literature but our preliminary normality tests did not justify its application. Given the extent of the collected samples, the Wilcoxon test was sufficient to support the conclusions reported. Besides, the Wilcoxon test is argued [25] to be statistically safer than the t-test for comparing classification algorithms over multiple data sets.

Acknowledgements
This article has been published as part of *BMC Bioinformatics* Volume 13 Supplement 10, 2012: "Selected articles from the 7th International Symposium on Bioinformatics Research and Applications (ISBRA'11)". The full contents of the supplement are available online at http://www.biomedcentral.com/bmcbioinformatics/supplements/13/S10.
This research was supported by the Czech Science Foundation through project No. 201/09/1665 (MH, FZ), the Czech Ministry of Education through research programme MSM 6840770012 (JK), and the Albert D. and Eva J. Corniea Chair for clinical research (JT).

Author details
[1]Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, 166 27, Czech Republic. [2]Department of Pediatrics, University of Minnesota, Minneapolis, 55454, USA.

Authors' contributions
MH collected the experimental data, implemented the experimental framework and accomplished the experiments. JK carried out the statistical evaluation of the study and partly wrote the manuscript. JK and FZ co-designed the experimental framework. FZ supervised all steps of the work and conceived the paper. JT motivated the initial phases of the study and revised the manuscript. All the authors read and approved the manuscript.

Competing interests
The authors declare that they have no competing interests.

Published: 25 June 2012

References

1. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gilette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, Mesirov JP: **Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles.** *PNAS* 2005, **102(43)**:15545-50.
2. Goeman JJ, Bühlmann P: **Analyzing gene expression data in terms of gene sets: methodological issues.** *Bioinformatics* 2007.
3. Dinu I: **Improving gene set analysis of microarray data by SAM-GS.** *BMC Bioinformatics* 2007.
4. Holec M, Zelezny F, Klema J, Tolar J: **Integrating Multiple-Platform Expression Data through Gene Set Features.** *The 5th International Symposium on Bioinformatics Research and Applications (ISBRA 2009)* Springer; 2009.
5. Mootha V, Lindgren C, *et al*: **SL: PGC-1-alpha-responsive genes involved in oxidative phosphorylation are coorinately down regulated in human diabetes.** *Nature Genetics* 2003, **34**:267-273.
6. Huang DWW, Sherman BTT, Lempicki RAA: **Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists.** *Nucleic acids research* 2008.
7. Tomfohr J, Lu J, Kepler TB: **Pathway level analysis of gene expression using singular value decomposition.** *BMC Bioinformatics* 2005, **6**:225.
8. Hastie T, Tibshirani R, Friedman J: *The Elements of Statistical Learning* Springer; 2001.
9. Golub TR, Slonim DK, Tamayo P, C Huard MG, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, Bloomfield CD, Lander ES: **Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring.** *Science* 1999, **286(5439)**:531-537.
10. Mitchell T: *Machine Learning* McGraw Hill; 1997.
11. Vapnik VN: *The Nature of Statistical Learning* Springer; 2000.
12. Gamberger D, Lavrac N, Zelezny F, Tolar J: **Induction of comprehensible models for gene expression datasets by subgroup discovery methodology.** *Journal of Biomedical Informatics* 2004, **34(4)**:269-284.
13. Zintzaras E, Kowald A: **Forest classification trees and forest support vector machines algorithms: Demonstration using microarray data.** *Cell Cycle* 2010, **40(5)**:519-24.
14. Huang J, Fang H, Tong W, X XF: **Decision forest for classification of gene expression data.** *Cell Cycle* 2010.
15. Liu J, Hughes-Oliver JM, Menius JA Jr: **Domain-enhanced analysis of microarray data using GO annotations.** *Bioinformatics* 2007, **23(10)**:1225-34.
16. Chen X, Wang L, Smith JD, Zhang B: **Supervised principal component analysis for gene set enrichment of microarray data with continuous or survival outcomes.** *Bioinformatics* 2008, **24(21)**:2474-81.
17. Guo Z, Zhang T, Li X, Wang Q, Xu J, Yu H, Zhu J, Wang H, Wang C, Topol EJ, Rao S: **Towards precise classification of cancers based on robust gene functional expression profiles.** *BMC Bioinformatics* 2005, **6**:58 +.
18. Bild AH, Yao G, Chang JT, Wang Q, Potti A, Chasse D, Joshi MB, Harpole D, Lancaster JM, Berchuck A, Olson JA, Marks JR, Dressman HK, West M, Nevins JR: **Oncogenic pathway signatures in human cancers as a guide to targeted therapies.** *Nature* 2005, **439(7074)**:353-357.
19. Wong DJ, Liu H, Ridky TW, Cassarino D, Segal E, Chang HY: **Module map of stem cell genes guides creation of epithelial cancer stem cells.** *Cell stem cell* 2008, **2(4)**:333-344.
20. Lee E, Chuang HYY, Kim JWW, Ideker T, Lee D: **Inferring pathway activity toward precise disease classification.** *PLoS computational biology* 2008, **4(11)**:e1000217+.
21. Abraham G, Kowalczyk A, Loi S, Haviv I, Zobel J: **Prediction of breast cancer prognosis using gene set statistics provides signature stability and biological context.** *BMC Bioinformatics* 2010, **11**:277+.
22. Mramor M, Toplak M, Leban G, Curk T, Demsar J, Zupan B: **On utility of gene set signatures in gene expression-based cancer class prediction.** *JMLR Workshop and Conference Proceedings Volume 8: Machine Learning in Systems Biology* 2010, 55-64.
23. Liu H, Motoda H: *Feature Selection for Knowledge Discovery and Data Mining* Kluwer; 1998.
24. Allison DB, Cui X, Page GP, Sabripour M: **Microarray data analysis: from disarray to consolidation and consensus.** *Nature reviews. Genetics* 2006, **7**:55-65.
25. Demšar J: **Statistical Comparisons of Classifiers over Multiple Data Sets.** *Journal of Machine Learning Research* 2006, **7**:1-30.
26. Guyon I, Weston J, Barnhill S, Vapnik V: **Gene Selection for Cancer Classification using Support Vector Machines.** *mlj* 2002, **46**:389-422.
27. Huang DW, Sherman BT, Lempick RA: **Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources.** *Nature Protocols* 2009, **4**:44-57.
28. Ho T: **The random subspace method for constructing decision forests.** *Transactions on Pattern Analysis and Machine Intelligence* 1997, **20(8)**:832-44.
29. Tarca AL, Draghici S, Khatri P, Hassan SS, Mittal P, Kim JS, Kim CJ, Kusanovic JP, Romero R: **A novel signaling pathway impact analysis.** *Bioinformatics* 2009, **25**:77-82.
30. Witten IH, Frank E: *Data Mining: Practical machine learning tools and techniques.* 2 edition. Morgan Kaufmann, San Francisco; 2005.
31. Laiho P, Kokko A, Vanharanta S, Salovaara R, Sammalkorpi H, Järvinen H, Mecklin JP, Karttunen TJ, Tuppurainen K, Davalos V, Schwartz S, Arango D, Mäkinen MJ, Aaltonen LA: **Serrated carcinomas form a subclass of colorectal cancer with distinct molecular basis.** *Oncogene* 2007, **26(2)**:312-20.
32. Armstrong SA, Staunton JE, Silverman LB, Pieters R, den Boer ML, Minden MD, Sallan SE, Lander ES, Golub TR, Korsmeyer SJ: **MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia.** *Nature Genetics* 2002, **30**:41-7[http://www.ncbi.nlm.nih.gov/pubmed/11731795].
33. Farmer P, Bonnefoi H, Becette V, Tubiana-Hulin M, Fumoleau P, Larsimont D, Macgrogan G, Bergh J, Cameron D, Goldstein D, Duss S, Nicoulaz AL, Brisken C, Fiche M, Delorenzi M, Iggo R: **Identification of molecular apocrine breast tumours by microarray analysis.** *Oncogene* 2005, **24(29)**:4660-71.
34. Cutcliffe C, Kersey D, Huang CC, Zeng Y, Walterhouse D, Perlman EJ: **Clear cell sarcoma of the kidney: up-regulation of neural markers with**

activation of the sonic hedgehog and Akt pathways. *Clinical cancer research : an official journal of the American Association for Cancer Research* 2005, **11(22)**:7986-94.
35. Burczynski ME, Peterson RL, Twine NC, Zuberek KA, Brodeur BJ, Casciotti L, Maganti V, Reddy PS, Strahs A, Immermann F, Spinelli W, Schwertschlag U, Slager AM, Cotreau MM, Dorner AJ: **Molecular classification of Crohn's disease and ulcerative colitis patients using transcriptional profiles in peripheral blood mononuclear cells.** *The Journal of molecular diagnostics : JMD* 2006, **8**:51-61.
36. Hippo Y, Taniguchi H, Tsutsumi S, Machida N, Chong JM, Fukayama M, Kodama T, Aburatani H: **Global Gene Expression Analysis of Gastric Cancer by Oligonucleotide Microarrays.** *Cancer Res* 2002, **62**:233-240.
37. Freije WA, Castro-Vargas FE, Fang Z, Horvath S, Cloughesy T, Liau LM, Mischel PS, Nelson SF: **Gene expression profiling of gliomas strongly predicts survival.** *Cancer research* 2004, **64(18)**:6503-10.
38. Sun L, Hui AM, Su Q, Vortmeyer A, Kotliarov Y, Pastorino S, Passaniti A, Menon J, Walling J, Bailey R, Rosenblum M, Mikkelsen T, Fine HA: **Neuronal and glioma-derived stem cell factor induces angiogenesis within the brain.** *Cancer cell* 2006, **9(4)**:287-300.
39. Bhattacharjee A, Richards WG, Staunton J, Li C, Monti S, Vasa P, Ladd C, Beheshti J, Bueno R, Gillette M, Loda M, Weber G, Mark EJ, Lander ES, Wong W, Johnson BE, Golub TR, Sugarbaker DJ, Meyerson M: **Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses.** *Proceedings of the National Academy of Sciences of the United States of America* 2001, **98(24)**:13790-1375.
40. Beer DG, Kardia SL, Huang CC, Giordano TJ, Levin AM, Misek DE, Lin L, Chen G, Gharib TG, Thomas DG, Lizyness ML, Kuick R, Hayasaka S, Taylor JM, Iannettoni MD, Orringer MB, Hanash S: **Gene-expression profiles predict survival of patients with lung adenocarcinoma.** *Nat Med* 2002, **8(8)**:816-824.
41. Spira A, Beane JE, Shah V, Steiling K, Liu G, Schembri F, Gilman S, Dumas YM, Calner P, Sebastiani P, Sridhar S, Beamis J, Lamb C, Anderson T, Gerry N, Keane J, Lenburg ME, Brody JS: **Airway epithelial gene expression in the diagnostic evaluation of smokers with suspect lung cancer.** *Nature medicine* 2007, **13(3)**:361-6.
42. Talantov D, Mazumder A, Yu JX, Briggs T, Jiang Y, Backus J, Atkins D, Wang Y: **Novel genes associated with malignant melanoma but not benign melanocytic lesions.** *Clinical cancer research : an official journal of the American Association for Cancer Research* 2005, **11(20)**:7234-42.
43. Scherzer CR, Eklund AC, Morse LJ, Liao Z, Locascio JJ, Fefer D, Schwarzschild MA, Schlossmacher MG, Hauser MA, Vance JM, Sudarsky LR, Standaert DG, Growdon JH, Jensen RV, Gullans SR: **Molecular markers of early Parkinson's disease based on gene expression in blood.** *Proceedings of the National Academy of Sciences of the United States of America* 2007, **104(3)**:955-60.
44. Dahia PLM, Ross KN, Wright ME, Hayashida CY, Santagata S, Barontini M, Kung AL, Sanso G, Powers JF, Tischler AS, Hodin R, Heitritter S, Moore F, Dluhy R, Sosa JA, Ocal IT, Benn DE, Marsh DJ, Robinson BG, Schneider K, Garber J, Arum SM, Korbonits M, Grossman A, Pigny P, Toledo SPA, Nosé V, Li C, Stiles CD: **A HIF1alpha regulatory loop links hypoxia and mitochondrial signals in pheochromocytomas.** *PLoS genetics* 2005, **1**:72-80.
45. Gordon GJ: **Transcriptional profiling of mesothelioma using microarrays.** *Lung cancer (Amsterdam, Netherlands)* 2005, **49(Suppl 1)**:S99-S103.
46. Libalova H, Dostal MPR Jr, Topinka J, Sram RJ: **Gene Expression Profiling in Blood of Asthmatic Children Living in Polluted Region of the Czech Republic (Project AIRGEN).** *10th International Conference on Environmental Mutagens* 2010.
47. Best CJM, Gillespie JW, Yi Y, Chandramouli GVR, Perlmutter MA, Gathright Y, Erickson HS, Georgevich L, Tangrea MA, Duray PH, González S, Velasco A, Linehan WM, Matusik RJ, Price DK, Figg WD, Emmert-Buck MR, Chuaqui RF: **Molecular alterations in primary prostate cancer after androgen ablation therapy.** *Clinical cancer research : an official journal of the American Association for Cancer Research* 2005, **11(19 Pt 1)**:6823-34.
48. Yoon SS, Segal NH, Park PJ, Detwiller KY, Fernando NT, Ryeom SW, Brennan MF, Singer S: **Angiogenic profile of soft tissue sarcomas based on analysis of circulating factors and microarray gene expression.** *The Journal of surgical research* 2006, **135(2)**:282-90.
49. Carolan BJ, Heguy A, Harvey BG, Leopold PL, Ferris B, Crystal RG: **Up-regulation of expression of the ubiquitin carboxyl-terminal hydrolase L1 gene in human airway epithelium of cigarette smokers.** *Cancer research* 2006, **66(22)**:10729-40.
50. Kuriakose MA, Chen WT, He ZM, Sikora AG, Zhang P, Zhang ZY, Qiu WL, Hsu DF, McMunn-Coffran C, Brown SM, Elango EM, Delacure MD, Chen FA: **Selection and validation of differentially expressed genes in head and neck cancer.** *Cellular and molecular life sciences : CMLS* 2004, **61(11)**:1372-83.
51. Gashaw I, Grümmer R, Klein-Hitpass L, Dushaj O, Bergmann M, Brehm R, Grobholz R, Kliesch S, Neuvians TP, Schmid KW, von Ostau C, Winterhager E: **Gene signatures of testicular seminoma with emphasis on expression of ets variant gene 4.** *Cellular and molecular life sciences : CMLS* 2005, **62(19-20)**:2359-68.

doi:10.1186/1471-2105-13-S10-S15
Cite this article as: Holec *et al.*: Comparative evaluation of set-level techniques in predictive classification of gene expression samples. *BMC Bioinformatics* 2012 **13**(Suppl 10):S15.

Empirical Evidence of the Applicability of Functional Clustering through Gene Expression Classification

Miloš Krejník and Jiří Kléma

Abstract—The availability of a great range of prior biological knowledge about the roles and functions of genes and gene-gene interactions allows us to simplify the analysis of gene expression data to make it more robust, compact, and interpretable. Here, we objectively analyze the applicability of functional clustering for the identification of groups of functionally related genes. The analysis is performed in terms of gene expression classification and uses predictive accuracy as an unbiased performance measure. Features of biological samples that originally corresponded to genes are replaced by features that correspond to the centroids of the gene clusters and are then used for classifier learning. Using 10 benchmark data sets, we demonstrate that functional clustering significantly outperforms random clustering without biological relevance. We also show that functional clustering performs comparably to gene expression clustering, which groups genes according to the similarity of their expression profiles. Finally, the suitability of functional clustering as a feature extraction technique is evaluated and discussed.

Index Terms—Biological prior knowledge, gene expression, gene set analysis, clustering, feature extraction, classification.

1 Introduction

Currently, there is a large range of bioinformatics tools that exploit *prior knowledge* of gene function. One important way to make use of this knowledge is through *functional clustering* (FC), which aims to group genes according to their functional similarities. The notion of functional similarity is based on the assumption that genes with related functional annotation records are functionally related to each other. Various approaches for FC are available [1], [2], [3], [4], [5]. The various approaches differ in their selection, heterogeneity, and amount of employed prior biological knowledge, their notion of similarity between genes and the type of clustering algorithm used. The corresponding tools vary in their availability and serviceability.

The most frequent application of FC is to simply break down a large gene list into a manageable number of functionally related groups for further efficient *interpretation*. The origin of the gene list is commonly high-throughput genomic, proteomic and bioinformatics scanning approaches (mostly expression microarrays) that enable the researcher to select interesting (typically differentially expressed) genes. Thus, the FC tools contribute to gene-annotation *enrichment analysis*. The functional gene clusters can then be used to control the subsequent experiments such that a gene cluster is given preference, e.g., if most of its gene members are associated with highly enriched annotation terms that are found in the traditional enrichment analysis of the total gene list. Khatri et al. [6] introduced the first tool for gene ontology functional analysis, the first discussion and comparison of various statistical functional analysis models is available in [7]. The detailed overviews of enrichment tools can be found in [8], [9].

However, functional annotations can also be employed in *classification* of gene expression (GE) data to obtain more interpretable, robust, and potentially accurate predictive models. Classification based on GE monitoring by DNA microarrays (often referred to as molecular classification) is a natural learning task with immediate practical uses. There have been several early success stories [10], [11], [12], followed by a large number of studies with the main goal of predicting cancer outcome (an overview is provided, e.g., in [13]). Recent surveys [14], [15] have demonstrated serious technical flaws in a large proportion of these studies, which were published in high-impact biomedical journals, and have found that most of the published results are overly optimistic. The routine application of GE classification is limited by frequent inaccuracies in the resulting classifiers and their inability to be understood by physicians. Molecular classifiers based solely on GE in most cases cannot be considered useful decision-making tools or decision-supporting tools.

Recent efforts in the field of molecular classification aim to employ additional information available for genes, proteins, and tissues that are being studied. They follow the major trend that is currently prevailing in the area of general GE data analysis. The analysis that was formerly aimed at identifying *individual genes* that are differentially expressed across sample classes [16] now focuses on identifying entire sets of genes with significantly different expression [17], [18], [19]. The genes share a set of characteristics that are defined by prior biological knowledge. The *set-level techniques* applied to GE classification develop new features that correspond to gene sets that represent pathways, their

• *The authors are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27, Prague 6, Czech Republic. E-mail: krejnmi1@fel.cvut.cz, klema@labe.felk.cvut.cz.*

Manuscript received 22 Mar. 2011; revised 12 Dec. 2011; accepted 29 Dec. 2011; published online 23 Jan. 2012.
For information on obtaining reprints of this article, please send E-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-2011-03-0069. Digital Object Identifier no. 10.1109/TCBB.2012.23.

subclusters or gene-ontology terms at various levels of generality [20], [21]. The authors of [22] propose a method that integrates a priori the knowledge of a gene network into a classification that results in classifiers with biological relevance, a good classification performance, and an improved interpretability of the results. Lee et al. [23] introduced the concept of condition-responsive genes (CORGs), which are the genes with the highest discriminative power in a pathway. The activity of a pathway is defined as a vector of CORG expression activity, and markers based on CORGs have been shown to improve the predictive results when compared with the random gene subset for a pathway. In [24], the authors compute the pathway activity score and the pathway consistency score. These two scores are then used as features for classifying phenotypes. The consistency score is defined using gene interaction networks.

In this paper, we propose the use of FC as *a feature extraction* tool for subsequent classification of GE samples. The main idea is to replace the sample features that originally corresponded to genes with a lower number of more robust, more interpretable features that correspond to the gene cluster centers. The dimensionality reduction of GE data by gene clustering with subsequent classification has already been proposed in [25]. The method is referred to as the prototype gene method, and the authors suggest that more accurate (and presumably more interpretable) classifiers can be created. However, this conclusion is only drawn from a two-data set experiment. The paper does not employ any prior knowledge regarding gene function (the authors suggest that it will be used in future works) and derives the k-mean clusters by the euclidean distance based on the GE profiles themselves.

This paper primarily addresses *the extent to which FC is useful in the analysis of GE data.* We assert that this question can only be partially answered when FC is applied within its traditional enrichment framework. In [2], the authors note that there is not a null hypothesis test to directly compare the quality of clustering algorithms. General remarks on the challenges of assessing the capabilities of any gene-set analysis method in real experiments can be found in [26], [17]. The common difficulty is that the ground truth is never known. The clustering outcome is therefore evaluated mainly in terms of its *interpretability* and in the scope of functional annotation data. Cluster compactness and stability are the most informative indirect measures of clustering outcome quality based on this point of view. The other common way of evaluating interpretability is purely subjective. Biomedical researchers interpret particular clusters, pick the most interesting clusters (those that can be given a plausible explanation) and compare them manually with other clusters derived from other bioinformatics tools. Although the comparison is convincing and the applicability of prior biological knowledge is broadly taken for granted, this method of evaluation leaves much room for subjective analysis. The author of [27], [28] summarizes the principal reasons for the demonstrable increase in the rate of false positive findings in research in general. It is also shown that the analysis of high-dimensional molecular data is increasingly affected by the risk for false positive conclusions.

This study considers another relevant criterion of clustering quality: *performance*. The performance criterion is orthogonal to the criteria of interpretability. It evaluates the clustering outcome in a wider context of GE data that underlie the creation of the gene list that is to be interpreted. Clearly, it is important that the clusters are interpretable, but they also need to prove meaningful in the original setting. The common method of performance evaluation is as follows: First, the gene clusters that are differentially expressed among the sample classes are identified. Then, the top-ranking clusters are interpreted, and it is demonstrated that their meaning is consistent with the definition of sample classes, which typically concern diseased and nondiseased individuals or different disease variants. This method of evaluation is as subjective as the interpretability evaluation mentioned above.

We propose the employment of an indirect, but entirely objective and impartial, method based on *predictive accuracy* (PA) to assess the performance of gene clustering approaches. The PA is estimated from the classification framework. The methodology that allows us to use PA to compare the efficiency of various types of gene clustering approaches is given briefly as follows: First, the involved genes whose expression levels are measured are clustered. Second, the features of the GE samples that originally corresponded to genes are replaced by features that correspond to the centroids of the clusters. Third, the classifiers, which are prescribed by formal models to determine the class of the new, unclassified samples, are learned, and their unbiased PA is estimated. Finally, the difference in the PA achieved for the various gene-clustering approaches is statistically evaluated. Note that the first two steps correspond to the procedure called feature extraction. The last two steps implement and evaluate the classification. Here, they serve to compare the different methods of feature extraction.

We assert that, there are two necessary conditions for applying FC to the analysis of GE data. First, the gene functional clusters need to perform better than random gene clusters (random clustering (RC) decomposes a gene list by disregarding any available information on the genes). If not, the functional clusters have no meaning for the data that created the gene list. Second, the gene functional clusters must achieve a performance comparable to that of the clusters that are based on gene expression profile similarity (the approach mentioned earlier in [25]). If not, there is a straightforward way to better cluster the genes without knowledge regarding their functionality. Consequently, the vague initial question is rephrased in terms of two technical hypotheses that compare the PA achieved by classifiers based on different types of gene clustering approaches: 1) FC leads to a better predictive performance than *random clustering* without knowledge of biological relevance; and 2) FC and *gene expression clustering* (GEC), which groups genes according to the similarity of their expression profiles, have equally predictive performances.

This study should not be taken as an effort to develop the most accurate molecular classifiers. It instead aims to provide a robust test of the hypotheses stated above regarding the applicability of prior biological knowledge for further processing and understanding of GE data. To demonstrate the direct performance of FC in feature extraction for further classification, two more comparisons are drawn. We compare the FC-based feature extraction to *feature selection* (FS) that chooses the most differentially

expressed genes and to the *fundamental treatment* that learns using all original data features.

The rest of this paper is organized as follows: Section 2 gives details on the FC, RC, and GEC algorithms. Section 3 describes the experimental protocol and provides and interprets the hypothesis test results. Section 4 discusses a few additional issues on the applicability of FC. Section 5 reviews the contributions of this study and outlines directions for future work.

2 Methods

This section reviews the differences among the gene clustering approaches (FC, RC, and GEC) implemented here. It also summarizes the prior biological knowledge that is used in FC.

2.1 Biological Prior Knowledge

In this paper, we define prior biological knowledge as any information that is not available in a GE data set but that is related to the genes contained in the data set. There is a rich body of knowledge available for genes including a short textual description of gene function, the cellular location, a bibliography, interaction partners and links with other genes, membership and role in pathways, referential sequences and many other pieces of information.

The way we apply the biological prior knowledge in functional clustering was mainly inspired by the popular "DAVID Gene Functional Clustering Tool" [2], which represents one of the most consistent efforts to fuse the available knowledge found in various biological annotation databases (14 annotation categories including Gene Ontology, KEGG Pathways, BioCarta Pathways, Swiss-Prot Keywords). Technically, the uniform list of annotation terms adopted from DAVID is applied to describe each gene. The background knowledge is represented as a binary gene-term matrix enable to cope with the many-to-many gene-to-term relationships that are found in functional annotation databases.

On the other hand, there are obvious limitations of such a representation. The annotation does not fuse all of the possible heterogeneous knowledge resources, and gene links or genomic sequences cannot fit this format. The binary resolution ignores variance in reliability of the individual annotation records, e.g., the Gene Ontology evidence codes (the computationally derived annotations are generally thought to be of lower quality than those inferred from direct experimental evidence [29]). Pathways are treated as gene sets, their network structure is not concerned.

Because we implemented the presented method in R, we use the annotation packages from the open source Bioconductor bioinformatics software [30]. In particular, we use two annotation packages: the Affymetrix HuGeneFL Genome Array annotation data (hu6800.db for the GPL80 platform) and the Affymetrix Human Genome U133 Set annotation data version (hgu133a.db for the GPL96 platform), which correspond to the microarray chips from the data sets used in the experiments. Last but not least, there is a technical limitation of functional clustering caused by the significant number of probes and genes without annotation. In the employed versions 2.5.0 (hu6800.db) and 2.4.5 (hgu133a.db) of the annotation packages, 23 percent, respectively, 43 percent of the probes remain unannotated and thus excluded from clustering.

2.2 Gene Similarity/Distance

The proper distance function is a keystone of any clustering algorithm. The gene distance grows with the dissimilarity of a gene pair, and the normalized distance is a real number from $\langle 0, 1\rangle$, where 0 is the identity and 1 indicates the maximum possible dissimilarity. The gene similarity is the complement of the distance function to 1. The simplest definition of gene distance is applied in RC, where a pair of genes is assigned a random distance value. In GEC, the euclidean distance is used. The euclidean distance of two genes, u and v, is defined as

$$d(u,v) = \sqrt{\sum_{i=1}^{n} (x_{iu} - x_{iv})^2}, \tag{1}$$

where n is the number of samples and x_{iu} is the expression value of the gene, u, in sample, i. In FC, the kappa similarity measure adopted from [31], [2] is used. The kappa of a gene pair is computed from the binary vectors of the annotation terms assigned to the genes (the term can be present or absent for the given gene). The kappa of two genes, u and v, is defined as

$$\kappa(u,v) = \frac{O_{uv} - A_{uv}}{1 - A_{uv}}, \tag{2}$$

where O_{uv} represents the observed cooccurrence and A_{uv} represents the chance cooccurrence. Let $\mathcal{T}$ be a set of observed annotation terms, and let C_{00} be the number of terms that occur in neither u nor v. Let C_{01} be the number of terms that occur in v, but not in u, and let C_{10} be the number of terms that occur in u, but not in v. Finally, let C_{11} be the number of terms that are observed in both u and v. Then, O_{uv} and A_{uv} are defined as

$$O_{uv} = \frac{C_{11} + C_{00}}{|\mathcal{T}|}, \tag{3}$$

and

$$A_{uv} = \frac{C_{*1}C_{1*} + C_{*0}C_{0*}}{|\mathcal{T}|^2}, \tag{4}$$

where $C_{*1} = C_{01} + C_{11}$, $C_{1*} = C_{10} + C_{11}$, $C_{*0} = C_{00} + C_{10}$, and $C_{0*} = C_{00} + C_{01}$.

2.3 Clustering Algorithms

Gene clusters can be found using the gene distance/similarity measures. This section briefly reviews the clustering algorithms used earlier in FC and GEC and explains the choice of clustering algorithms made in this study.

The contribution of gene functional annotations in GE data analysis can be most easily illustrated when an identical clustering algorithm is used for functional, random, and gene expression clustering. By applying only one clustering algorithm, we can increase the reliability of the hypothesis tests, as the issue of the influence of the clustering algorithm and its parameterization on the PA can be completely omitted. Therefore, we have reviewed the

clustering algorithms that were actually applied earlier in FC and GEC, studied their evaluation or reevaluated them and attempted to identify an algorithm that best fits both fields of application. The algorithm selected also needs to be computationally feasible for large, genome-wide lists. Finally, the repetitive nature of our study needs to be addressed. In GEC, clustering needs to be performed for every single cross-validation split (10,000 total runs as we deal with 10 data sets, 10 fold cross validation, 10 numbers of clusters, and 10 repetitions). In FC, only 200 runs are needed (two platforms, 10 numbers of clusters, and 10 repetitions) because the clustering is independent of the GE data. Section 3.2 discusses the experimental design in detail.

The first candidate is the heuristic fuzzy partition (HFP) clustering algorithm that was developed for the DAVID Functional Annotation Clustering Tool [2]. The authors of the tool experimentally verified that fuzzy clustering best fits the gene annotation data and the nature of functional relationships of the genes from the viewpoint of interpretability. We therefore reimplemented the HFP clustering algorithm in R [32], accelerated it, and made it scalable to genome-wide experiments. However, we have found that the HFP clustering algorithm does not suit the gene profile similarities that have distributions that are unlike the kappa similarity distribution for functional annotations. The algorithm is difficult to regulate to obtain a reasonable number of reasonably sized clusters (small changes in the control parameters often result in very different clustering of the initial gene set). In addition, the HFP clustering algorithm has a higher empirical computational complexity than a crisp clustering algorithm such as k-means or k-medoids clustering, and applying it multiple times for GEC is not computationally feasible.

The second candidate for a uniform clustering algorithm is the k-means algorithm [33], which was applied for GEC in [25], [34], [35], [36], [37]. The algorithm appears to be suitable for the GEC application from the viewpoint of PA, its ease of control and its efficiency for repetitive execution. Although the algorithm cannot be immediately applied to FC because it deals with cluster centroids whose functional annotation vectors are unclear, it can be replaced by a similar algorithm: k-medoids [38]. We believe that the k-medoids algorithm is the best choice of the three for the following reasons: 1) the algorithm shares its main characteristics with the k-means algorithm; both of the algorithms are partitional, crisp (not fuzzy), and minimize the distance between objects that belong to a cluster and the center of that cluster; 2) as with the HFP clustering algorithm, the k-medoids algorithm uses medoids as cluster centers in the place of centroids; it also allows the use of a similarity matrix instead of the data matrix for the input (the object coordinates in the feature space do not need to be available), and it is therefore more suitable for use with the κ similarity that is recommended by the DAVID Functional Annotation Clustering Tool; and 3) although fuzziness is a desirable property because of the biological nature of the gene functions and the resulting enhanced capabilities, e.g., for interpretation of the results, we have experimentally verified that the impact of k-medoids on PA with respect to the HFP clustering algorithm is marginal and appears to be positive.

In the end, our study implements two different clustering algorithms. We used our own Python implementation of the k-medoids algorithm for FC, whereas GEC employs a Scipy [39] implementation of the k-means algorithm as a benchmark algorithm for GEC. In both FC and GEC, the initial medoids and centroids, respectively, were selected randomly from the considered genes. RC starts with the gene clusters that are found by FC. Then, the genes are randomly shuffled among the clusters. The random shuffling preserves the cluster sizes found in the FC and guarantees that the differences between the RC and FC are not the result of a different number and size of the clusters.

We believe that this strategy results in a less biased analysis than the direct comparison between the most frequently used algorithms for FC and GEC, which are the HFP and k-means algorithms. We have experimentally verified that the answers for the key questions remain the same with regard to FC and RC, which are driven by the HFP clustering algorithm and GEC performed with the k-means clustering algorithm. In this study, we emphasize the hypothesis tests that are reached with similar clustering algorithms in FC, RC, and GEC, as they allow for a simpler and more readable formulation of the second technical hypothesis.

2.4 Cluster Expressions

After gene clustering, the expression of the gene clusters needs to be computed. The original GE data sets are transformed from the original m-dimensional gene space into q-dimensional cluster space ($m \gg q$). Let $\mathbf{x}_i = (x_{i1}, \ldots, x_{im})$ be a sample from the original feature space, where x_{ij}, $j = 1, \ldots, m$ is the expression value of the gene, j, in the sample, i. Next, let $C_1, \ldots, C_q$ be the gene clusters found via a particular clustering algorithm. Then, $\tilde{\mathbf{x}}_i = (\tilde{x}_{i1}, \ldots, \tilde{x}_{iq})$ is a sample from the q-dimensional reduced space, where $\tilde{x}_{ij}$, $j = 1, \ldots, q$ is the expression for the value of the gene cluster, j, in the sample, i, which is computed as

$$\tilde{x}_{ij} = \frac{\sum_{g \in C_j} x_{ig}}{|C_j|}. \tag{5}$$

3 Experiments

The goal of the conducted experiments was to compare FC, RC, and GEC in terms of the PA of the classifiers learned on data sets that have the dimensions reduced by the given gene clustering approach. In this section, we describe the data sets that were used as well as the experimental framework, and we summarize the results.

3.1 Data Sets

For the experiments, we used a set of 10 publicly available GE data sets that have two class labels. The key parameters of the data sets are summarized in Table 1. The data sets were normalized by quantile normalization [49] to have the same distribution of GE for each sample in the given data set. The following criteria were considered during the data sets selection:

TABLE 1
An Overview of the Key Parameters of the Benchmark Data Sets

Dataset	Reference	Number of samples	Class ratio	Number of features	Platform
ALL/AML	[12]	72	47:25	7,129	GPL80
AML	[40]	64	38:26	22,283	GPL96
Breast cancer	[41]	29	15:14	22,283	GPL96
Gastric cancer	[42]	30	22:8	7,129	GPL80
Glioma	[43]	85	59:26	22,283	GPL96
Hypertension	[44]	20	14:6	7,129	GPL80
MGCT	[45]	27	18:9	22,283	GPL96
Prostate cancer	[46]	20	10:10	22,283	GPL96
Sarcoma/Hypoxia	[47]	54	39:15	22,283	GPL96
Smoking	[48]	44	26:18	7,129	GPL80

1. availability—all of the data sets are publicly available via NCBI GEO [50] and have preferably been used by other researchers as benchmarks;
2. informedness—the GE measured must correlate with the target class somehow; otherwise, no clustering or learning approach will differ from random assortment;
3. difficulty—the relationship between GE and the target class must not be trivial or absolute; if a single gene perfectly splits the samples then there is no room for gene clustering; and
4. platform—we deal with only two microarray platforms to accelerate the experiments (RC and FC remain identical for different data sets that use the same platform).

3.2 Design

To compare FC, RC, and GEC, we used 10 k values ($k = 2^c, c = 1, 2, \ldots, 10$) that determine the number of clusters, 10 data sets (see Section 3.1) and five classification algorithms (see below). For each combination of the gene clustering approach, number of clusters, classification algorithm, and data set, a PA value is computed as follows: At first, 10 partial PA values are computed, each of them is computed via stratified 10-fold cross validation (as recommended in [51]) with different random seeds for the cluster initialization. Then, the final PA for the given combination is computed as the average of 10 partial PA values. The partial PA values are computed and averaged to avoid bias from random shuffling in RC and from random initialization in FC and GEC. In this way, 500 (10 k values $\times$ 10 data sets $\times$ 5 classification algorithms) final values of the PA for each gene clustering approach are obtained.

The particular classifiers were learned by five different classification algorithms: support vector machines (SVM) [52] (with linear kernel and hyperparameters of $C = 1.0$ and $\epsilon = 0.1$), random forests (RF) [53] (with 100 random trees from $\sqrt{n}$ random features, where n is the size of original dimension), C4.5 [54], naïve Bayes (NB) [55], and nearest neighbor (NN) [56]. The support vector machines represent the most frequently used classification algorithm in GE classification [57]. They are known to be able to cope with unfavorable rates of sampling (tissues and other biological situations) and variables (features or genes). The random forests method represents a robust ensemble classification algorithm that is suitable for GE data [58], whereas C4.5 produces decision trees that are instantly readable by a human and are the first option based on interpretability. The naïve Bayes and nearest neighbor algorithms represent classic and computationally efficient classification algorithms that are known to have reasonable accuracy, and in our study, they primarily serve to minimize the learning bias.

We opted for the modifications of the classification algorithms that require as few hyperparameters as possible to avoid needing another nested cross-validation cycle to optimize them. The nested cross-validation is time consuming, especially for GEC, as it would multiply the number of clustering runs (genome-wide clustering is the most time-consuming step). It also tends to decrease the sample numbers and the variability in the individual stratified folds. The actual applied hyperparameters are known to be robust at their default setting (support vector machines) or there has been a recommendation for their heuristic prior initialization (random forests). Orange [59] implementation of the classification algorithms was applied.

3.3 Results

By applying the described procedure, 1,500 (3 gene clustering approaches $\times$ 10 k values $\times$ 10 data sets $\times$ 5 classification algorithms) estimations of PA were obtained. The main objective of our study is to compare the individual gene clustering approaches. The hypotheses regarding the equality of the gene clustering approaches in terms of their predictive performance were tested via the Wilcoxon signed-rank test [60], as recommended in [61] in place of the widely used t-test. The hypotheses were tested at a level of significance of $\alpha = 0.05$. If not stated otherwise, the same statistical test and the same α level were used in other experiments too.

First, the medians over the 500 PA values available for the individual clustering approaches can be computed. However, this condensed summary gives only a rough view of the total performance because the PA measured in the different domains is not commensurable and is highly variable; therefore, aggregating it over domains is not meaningful [61]. Instead, mutual direct comparisons should be based on the gene clustering approach rankings, which consider successes and failures rather than the absolute

TABLE 2
Mean Ranks of the Gene Clustering Approaches with Regard to PA

Dataset	FC	GEC	RC
ALL/AML	1.53	1.82	2.64
AML	2.22	1.58	2.20
Breast cancer	2.00	2.10	1.90
Gastric cancer	1.52	2.26	2.21
Glioma	2.32	1.62	2.05
Hypertension	1.50	2.42	2.07
MGCT	2.33	1.46	2.21
Prostate cancer	1.91	1.82	2.27
Sarcoma/Hypoxia	2.00	1.26	2.74
Smoking	1.20	2.98	1.82
All	1.85	1.93	2.21

The table shows the mean domain ranks (averaged over all of the classification algorithms and k values) and the total mean ranks (averaged over all of the domains, last row).

accuracy of the methodology. For example, for the ALL/AML domain, naïve Bayes classifier algorithm and $k = 16$ (16 clusters), the gene clustering approaches had accuracies of FC 90 percent, RC 83 percent, and GEC 92 percent. The ranking is FC—2nd, RC—3rd and GEC —1st; the difference in the PA does not matter. The mean ranks are meaningful even if they are obtained over different data sets. The conclusions with regard to the ranks of the clustering approaches are shown in Table 2 (the final row gives a condensed summary). As outlined in Section 1, our main interest is in paired FC versus RC and in FC versus GEC. The first null hypothesis, that FC and RC have equally predictive performances, was rejected in favor of the alternative hypothesis, FC has a higher predictive performance than RC (one-sided test, p-value = 0.042, which is $<\alpha$). The second null hypothesis, FC and GEC are equally predictive, could not be rejected in favor of the alternative hypothesis, FC and GEC have distinct predictive performances (two-sided test, p-value = 0.85, which is $>\alpha$).

However, the most relevant conclusions must be drawn from the paired differential analysis that has the largest statistical power. The analysis relates the accuracy values reached by two gene-clustering approaches when the other settings are identical. Our interest is again in paired FC versus RC and in FC versus GEC; therefore, 500 (10 k values $\times$ 10 data sets $\times$ 5 classification algorithms) differential values are obtained for each pair when the differential accuracy for both of the clustering pairs is calculated. The box plots for the particular data sets and the clustering pairs are depicted in Fig. 1.

The following statistical test summarizes the visual differences seen in the results shown in Fig. 1. Prior to the test, the aggregate across the k values and classification algorithms has to be calculated because the runs with different classification algorithms and different k values within a data set are dependent (that is, a higher accuracy in one predicts a higher accuracy in the others and the same holds true for differences). Then, the final test deals with 10 medians of 50 (10 k values $\times$ 5 classification algorithms) different accuracy values. In other words, it tests a vector of 10 independent median values that are derived for 10 different data sets. The median is used in place of the mean because the differential accuracy for the particular data sets has an asymmetric distribution.

The first null hypothesis, that FC and RC have equally predictive performances, was rejected in favor of the alternative hypothesis, FC has a higher predictive performance than RC (one-sided test, p-value = 0.019, which is $<\alpha$). FC performed better than RC on eight out of 10 benchmark data sets. Compared with a randomly selected gene set, the functional cluster has increased interpretability and performance.

The second null hypothesis, FC and GEC are equally predictive, could not be rejected in favor of the alternative hypothesis, FC and GEC have distinct predictive performances (two-sided test, p-value = 0.92, which is $>\alpha$). FC performed better on five of 10 benchmark data sets, and GEC performed better on the other five data sets, which suggests that functional clusters represent an alternative to

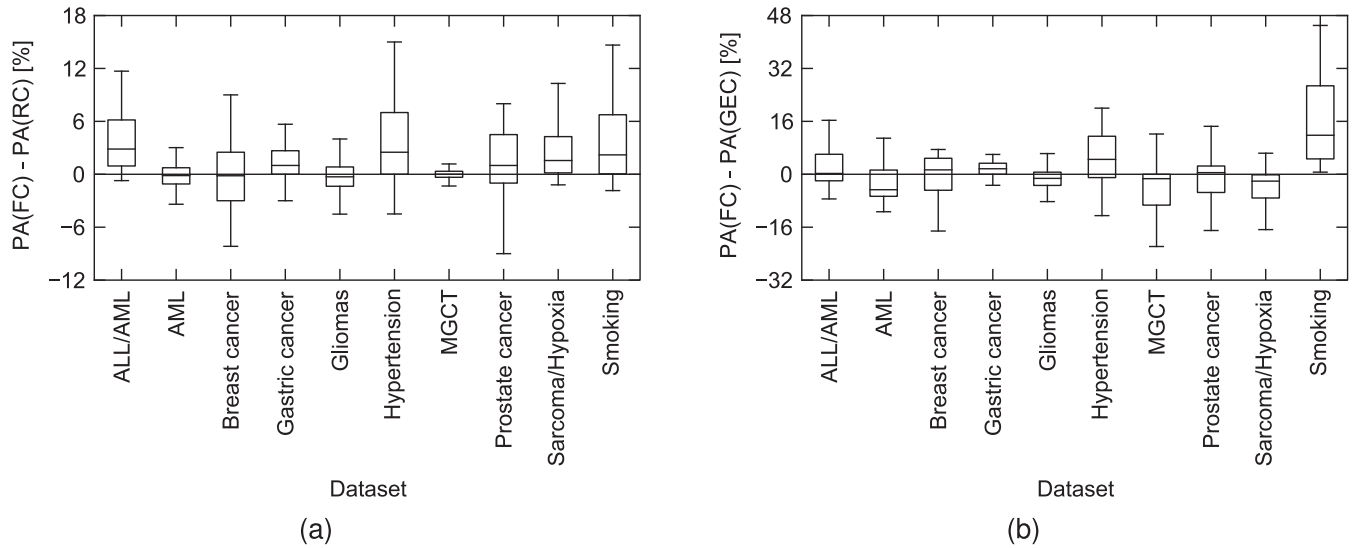


Fig. 1. Box plots for the PA differences for the given data sets and hypotheses: (a) FC versus RC; and (b) FC versus GEC. Each box plot is computed from 50 (10 k values $\times$ 5 classification algorithms) values for the PA difference.

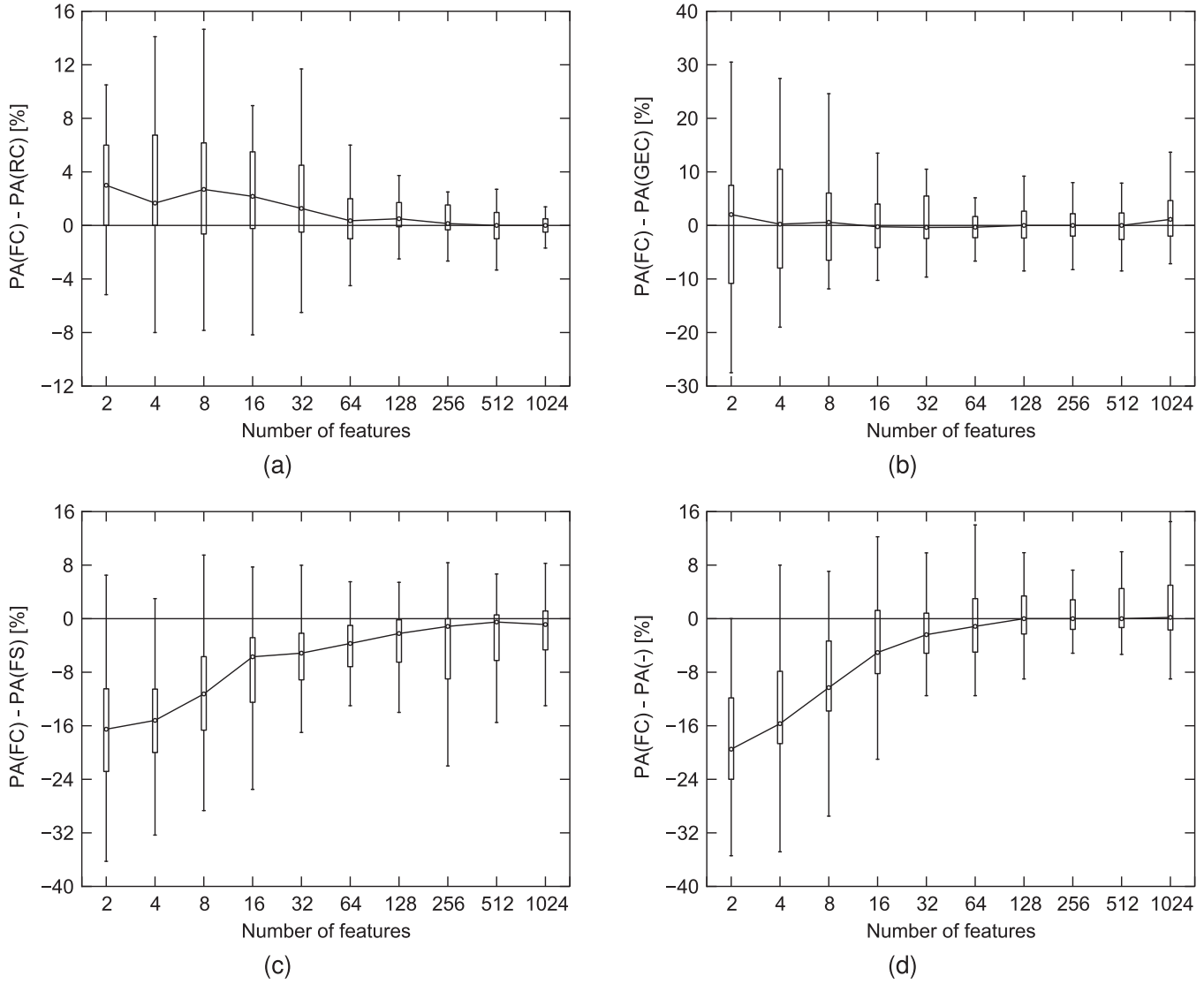


Fig. 2. Box plots for the PA differences for a given number of features and pairs of feature extraction/selection approaches: (a) FC versus RC; (b) FC versus GEC; (c) FC versus FS; and (d) FC versus the full gene set without dimension reduction. Each box plot is computed from 50 (10 data sets $\times$ 5 classification algorithms) values for the PA difference.

purely statistical clusters in terms of PA. Note that GEC often identifies gene clusters that share no common annotation pattern and cannot be plainly interpreted. In the case of equally predictive performances, preference is given to the more interpretable option. This option is clearly represented by functional clusters, which are naturally complemented by a shared functional pattern.

4 Discussion

This section provides comments that will aid in the interpretation of the results provided in the previous section, describes the influence from the number of clusters and the classification algorithm and compares two principal approaches for dimensionality reduction. Although the discussed issues can be regarded as technical details with respect to the key questions, they may help place the results into perspective and provide additional details.

4.1 Number of Clusters

The clustering algorithms used enabled us to immediately compare the gene clustering approaches based on the functional, gene-expression-based, and random gene distances across the considered k values. The differential comparisons can be seen in Fig. 2. The margin between FC and RC is most distinct for lower numbers of clusters and tends to decrease steadily as the number of clusters increase. A few large random clusters have significantly less information than the functional ones, whereas the large number of smaller random clusters can have nearly the same level of informedness as the functional ones. This observation is in agreement with an earlier conclusion that the enrichment of gene expression clusters for biological function is generally the highest at a relatively low number of clusters [62]. FC generates large clusters of genes that tend to share expression profiles, and this relationship decreases as the number of clusters increases. The margin between functional clustering and GEC does not show a strong pattern.

Fig. 3 shows that the PA increases with an increasing number of clusters. The gene clustering approaches are comparable with the full set of features (the dotted line) when the number of clusters reaches approximately 100, which suggests that the original performance can be maintained with a reasonable dimensionality reduction;

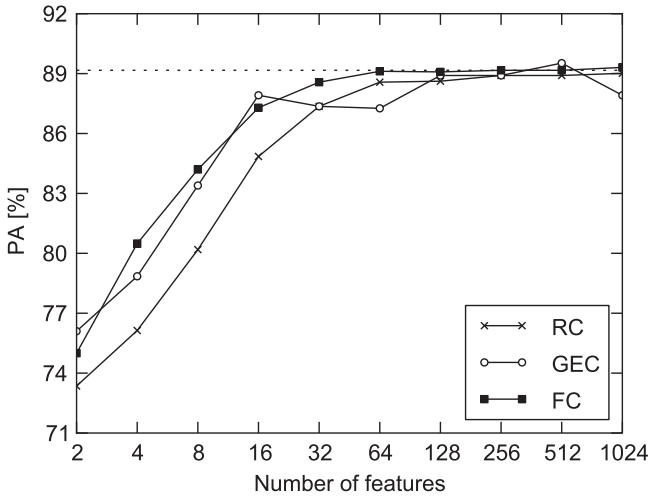


Fig. 3. Medians of the PAs for three ways of gene clustering. The dotted line represents the median of the PAs for the full gene set without dimension reduction. Each median is computed from 50 (10 data sets $\times$ 5 classification algorithms) values for the PA.

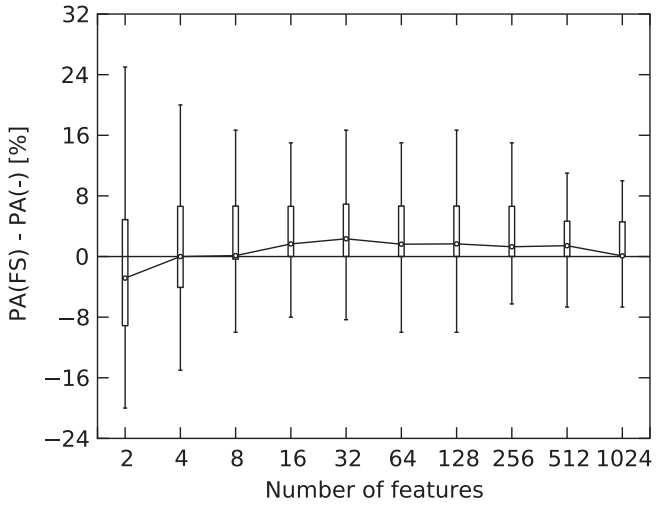


Fig. 5. Differential PA box plots comparing classification based on FS and the full gene set without dimension reduction. Each box plot is computed from 50 (10 data sets $\times$ 5 classification algorithms) PA differential values.

however, the number of clusters cannot be extremely low without sacrificing PA. Note that the optimal number of clusters differs across domains. As a matter of fact, there are five domains with a clear coherent range of the numbers of clusters with the PA of FC higher than the referential one derived from the full gene set. This characteristic is not obvious in Fig. 3 for its aggregation over domains.

4.2 Classification Algorithms

We experimented with five diverse classification algorithms (see Section 3.2). None of the methods given below is superior to the others in principle. The main reason for using the pool of learning algorithms is to avoid a dependence of the experimental results on their specific biases. Therefore, the answer given by the pool of methods is more illustrative and robust than the answers provided by any given method. Still, a brief comparison of the classification algorithms can illustrate their differences. Fig. 4 shows the overall performance of the individual algorithms. The

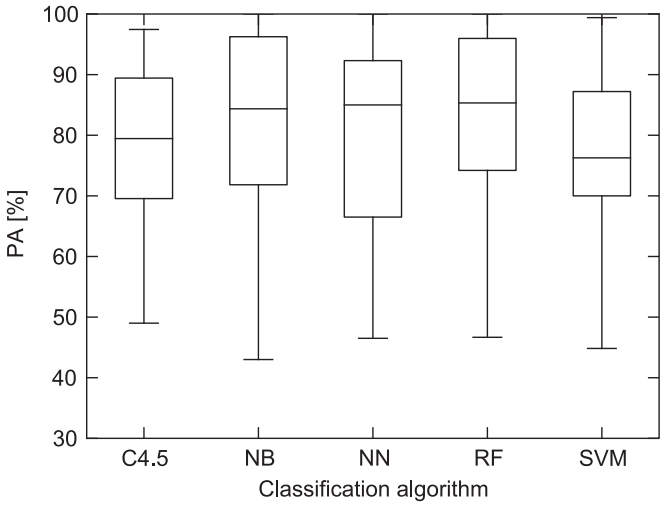


Fig. 4. Box plots for the PAs for the given classification algorithms, namely C4.5, naïve Bayes, nearest neighbor, random forests, and support vector machines. Each box plot is computed from 300 (three gene clustering approaches $\times$ 10 k values $\times$ 10 data sets) values for the PA.

only significantly different pairs are random forests versus C4.5 and random forests versus support vector machines (Friedman test [63], p-value = 0.019 and p-value = 0.039, respectively). The low accuracy of the support vector machines algorithm (with a linear kernel) indicates the nonlinearity of the classification problems that are being considered. The improved accuracy of FC with respect to RC is preserved across the classification algorithms; its significance can be proven for the nearest neighbor and random forests algorithms (one-sided test with Bonferroni-Dunn adjustment, p-value = 0.003 and p-value = 0.041, respectively).

4.3 Feature Selection

This paper focuses on clustering as a method that reduces the dimensionality of GE data. The new features that are generated are represented by the cluster centroids, which are extracted from the original features. The parallel approach to dimensionality reduction lies in feature selection; a review of its use in bioinformatics can be found in [64]. FS is frequently implemented with GE data for the selection of differentially expressed genes. Criteria such as the absolute t-test statistic can be used to rank the genes, and permutation tests can help to establish a threshold for genes that are significantly related to the response. To place the algorithms for feature extraction that were discussed and compared in this study into a wider context, we also compared their performance against FS. We ranked the genes by t-test, selected the most differentially expressed genes (the thresholds were gradually set to match the number of clusters) and ran the classification algorithms. The process was repeated 10 times for 10-fold cross validation. As shown in Fig. 2c, the PA achieved is clearly superior to that achieved by clustering. The null hypothesis that FS and FC have equally predictive performances was rejected (two-sided test, p-value = 0.002), which is not surprising because FC ignores the sample class labels, a significant information source for the feature transformation phase. Fig. 5 demonstrates that FS improves a PA in comparison with the full gene set without dimension reduction.

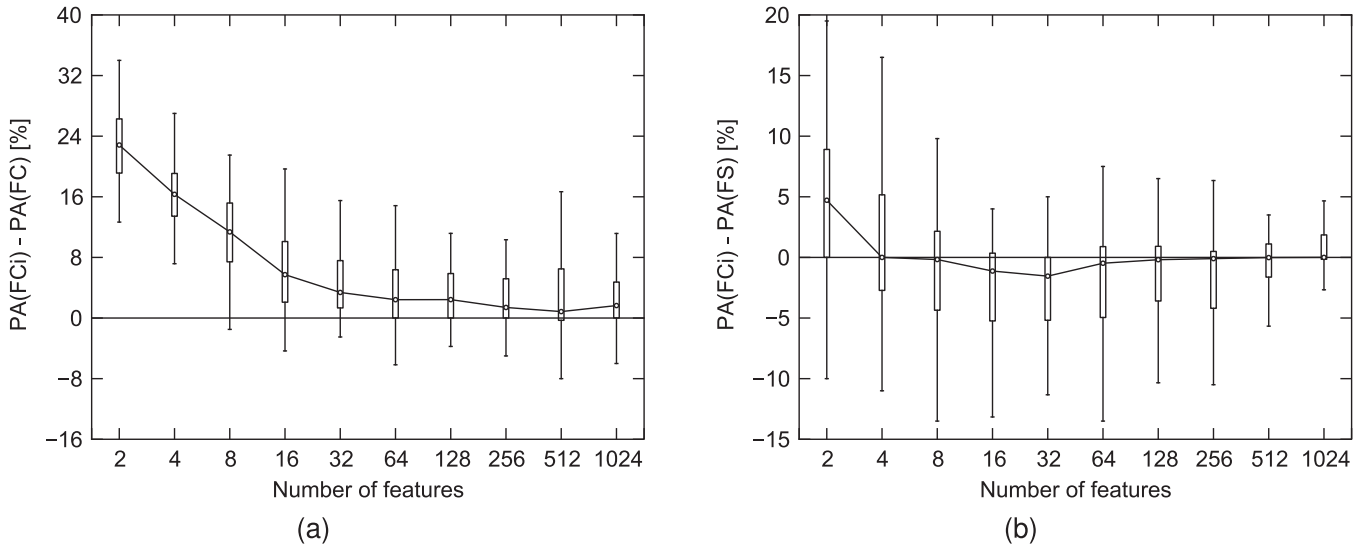


Fig. 6. Illustration of the effects of FC improvements. Box plots for the PA differences for a given number of features and pairs of feature extraction/selection approaches: (a) FCi versus FC; (b) FCi versus FS. Each box plot is computed from 50 (10 data sets $\times$ 5 classification algorithms) values for the PA difference.

4.4 Functional Clustering Improvements

Our study did not aim to achieve the maximum PA. To do so, FS would clearly be the first dimension reduction option chosen on the basis of its simplicity and performance. Maximization of the PA by FC would include FS as one of the early steps. We have implemented and tested a simple FC improvement (FCi) that exploits FS and the sample class labels: 1) in order to reduce noise, the cluster centroids represent only differentially expressed probes (t-test is applied, the probes with p-value $< 0.01 \log_2 k$ are used, the threshold increases with k to minimize empty or trivial centroids); 2) in order to minimize the negative influence of averaging, each cluster is represented by two centroids, upregulated and downregulated probes are treated separately; and 3) to keep the number of centroids equal with the number of clusters, the final set of k cluster centroids is made by the most differentially expressed ones. Fig. 6a shows that the improvements boost the PA of FC, Fig. 6b demonstrates that its performance becomes comparable with FS. Although it can be argued that FS is still an easier method to reduce dimension, the above described experiment suggests that the approaches that combine FC with FS (and potentially GEC) shall not be ignored.

5 Conclusion

This paper proposes a general methodology to impartially verify the applicability of particular types of gene clustering approaches. The verification is conducted within the predictive classification framework and focuses on prior biological knowledge-based FC. The framework uses three parallel methods of gene clustering. It statistically tests for differences in the PA of machine learning classifiers that are trained on the centroids of particular clusters. We experimentally verified that FC has a higher PA than RC without biological relevance. The effect of prior biological knowledge is remarkable for two main reasons: 1) it can be statistically verified for a limited set of 10 GE data sets; and 2) it persists in simplified cluster construction based on GE averaging (see (5)), which does not distinguish between gene activation and inhibition. We also showed that FC performs comparably to GEC, which groups genes according to the similarity of their expression profiles.

In addition, we showed that FC can provide a reasonable dimensionality reduction without sacrificing the PA achieved with the full set of features. This observation is promising concerning simplicity of the currently implemented FC, namely the above-mentioned cluster averaging, but also the frequent utilization of genes whose GE profiles have no relation to the phenotype, the imperfections in gene distance calculation and the probes and genes with missing annotations. Another interesting characteristic is that FC is carried out independently of GE data, which makes it an unsupervised and potentially computationally efficient feature extraction technique. Unlike GEC, FC is carried out just once per a particular gene set (platform) and the clusters are immediately applicable across the GE experiments using the particular platform.

At the same time, it holds that FC does not achieve a PA that is comparable to that achieved by FS, and combining the two techniques would maximize performance. It was experimentally demonstrated that FS is a simple method that improves a PA in a vast majority of domains (of course, the conclusion is influenced by the selection of classification algorithms and their noise robustness) and differential expression can hardly be ignored when calculating the cluster aggregates.

There are several directions for future work. First, the current pair of hypotheses can logically be supplemented by a third null hypothesis, there is no synergic action between the knowledge-based FC and GE-based GEC. We showed that both GE data and prior biological knowledge regarding gene roles, functions and interactions can underlie the creation of gene clusters. There are at least three reasons to believe that these algorithms can complement each other: 1) FC corresponds to a universal gene partitioning, whereas GEC provides a local partitioning for specific biological conditions; 2) FC clusters only the genes with an existing annotation, whereas genes without an annotation are left unused or create a cluster without real meaning; GEC uses all of the genes (both with and without annotation), which gives GEC an advantage over FC; and

3) FC deals with human-created annotations, whereas GEC creates ad hoc links based on a limited number of arrays that are known to provide only a noisy image of gene actions. However, the testing of this hypothesis lies beyond the scope of this paper, as there are many ways to aggregate clusters raised from FC and GEC into unified knowledge and statistical groups. Some general ideas regarding clustering aggregation can be found in [65]. In [66], the authors introduce the problem of combining multiple partitions of a set of objects into a single consolidated clustering without accessing the features or algorithms that determined these partitions. A discussion on early, intermediate, and late integration of microarray and medical literature data for gene clustering can be found in [67].

Second, the current cluster expression is computed in the most straightforward way by averaging the expression levels of the cluster members. A more complex cluster activity function could also consider the internal structure of the gene set that generates a cluster. The structure could potentially be extracted from the prior biological knowledge, and it could also be (re)invented statistically from GE data. However, preliminary efforts to employ the statistical SVD method for constructing metagenes proposed in [68] did not provide a detectable immediate improvement [21].

ACKNOWLEDGMENTS

The work of Miloš Krejník was funded by the Grant Agency of the Czech Technical University in Prague (grant no. SGS10/187/OHK3/2T/13). The work of Jiří Kléma was funded by the Czech Ministry of Education in the framework of the research program, Transdisciplinary Research in the Area of Biomedical Engineering II (MSM 6840770012). The authors thank Tomáš Sixta for reimplementation of the DAVID clustering algorithm in R. They also thank Robin Healey for English proofreading.

REFERENCES

[1] D. Chaussabel and A. Sher, "Mining Microarray Expression Data by Literature Profiling," *Genome Biology,* vol. 3, no. research0055, 2002.

[2] D.W. Huang, B.T. Sherman, Q. Tan, J.R. Collins, W.G. Alvord, J. Roayaei, R. Stephens, M.W. Baseler, H.C. Lane, and R.A. Lempicki, "The David Gene Functional Classification Tool: A Novel Biological Module-Centric Algorithm to Functionally Analyze Large Gene Lists," *Genome Biology,* vol. 8, no. R183, 2007.

[3] J. Natarajan and J. Ganapathy, "Functional Gene Clustering via Gene Annotation Sentences, MeSH and GO Keywords from Biomedical Literature," *Bioinformation,* vol. 2, no. 5, pp. 185-193, 2007.

[4] K. Ovaska, M. Laakso, and S. Hautaniemi, "Fast Gene Ontology Based Clustering for Microarray Experiments," *BioData Mining,* vol. 1, no. 11, 2008.

[5] G. Macintyre, J. Bailey, D. Gustafsson, I. Haviv, and A. Kowalczyk, "Using Gene Ontology Annotations in Exploratory Microarray Clustering to Understand Cancer Etiology," *Biochemistry,* vol. 31, no. 14, pp. 2138-2146, 2010.

[6] P. Khatri, S. Draghici, G.C. Ostermeier, and S.A. Krawetz, "Profiling Gene Expression Using Onto-Express," *Genomics,* vol. 79, no. 2, pp. 266-270, 2002.

[7] S. Draghici, P. Khatri, R. Martins, G. Ostermeier, and S. Krawetz, "Global Functional Profiling of Gene Expression," *Genomics,* vol. 81, no. 2, pp. 98-104, 2003.

[8] P. Khatri and S. Draghici, "Ontological Analysis of Gene Expression Data: Current Tools, Limitations, and Open Problems," *Bioinformatics,* vol. 21, no. 18, pp. 3587-3595, 2005.

[9] D.W.W. Huang, B.T.T. Sherman, and R.A.A. Lempicki, "Bioinformatics Enrichment Tools: Paths Toward the Comprehensive Functional Analysis of Large Gene Lists," *Nucleic Acids Research,* vol. 37, no. 1, Nov. 2008.

[10] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini, "Tissue Classification with Gene Expression Profiles," *Proc. Fourth Ann. Int'l Conf. Computational Molecular Biology,* pp. 54-64, 2000.

[11] S. Dudoit, J. Fridlyand, and T.P. Speed, "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data," *J. Am. Statistical Assoc.,* vol. 97, no. 457, pp. 77-87, 2002.

[12] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science,* vol. 286, no. 5439, pp. 531-537, 1999.

[13] J. Lee, J. Lee, M. Park, and S. Song, "An Extensive Evaluation of Recent Classification Tools Applied to Microarray Data," *Computational Statistics and Data Analysis,* vol. 48, no. 4, pp. 869-885, 2005.

[14] A. Dupuy and R. Simon, "Critical Review of Published Microarray Studies for Cancer Outcome and Guidelines on Statistical Analysis and Reporting," *J. Nat'l Cancer Institute,* vol. 99, no. 2, pp. 147-157, 2007.

[15] S. Michiels, S. Koscielny, and C. Hill, "Prediction of Cancer Outcome with Microarrays: A Multiple Random Validation Strategy," *The Lancet,* vol. 365, no. 9458, pp. 488-492, 2005.

[16] V.G. Tusher, R. Tibshirani, and G. Chu, "Significance Analysis of Microarrays Applied to the Ionizing Radiation Response," *Proc. Nat'l Academy of Sciences USA,* vol. 98, no. 9, pp. 5116-5121, 2001.

[17] A. Subramanian, P. Tamayo, V.K. Mootha, S. Mukherjee, B.L. Ebert, M.A. Gillette, A. Paulovich, S.L. Pomeroy, T.R. Golub, E.S. Lander, and J.P. Mesirov, "Gene Set Enrichment Analysis: A Knowledge-Based Approach for Interpreting Genome-Wide Expression Profiles," *Proc. Nat'l Academy of Sciences USA,* vol. 102, no. 43, pp. 15545-15550, 2005.

[18] I. Dinu, J. Potter, T. Mueller, Q. Liu, A. Adewale, G. Jhangri, G. Einecke, K. Famulski, P. Halloran, and Y. Yasui, "Improving Gene Set Analysis of Microarray Data by SAM-GS," *BMC Bioinformatics,* vol. 8, no. 242, 2007.

[19] Y. Hippo, H. Taniguchi, S. Tsutsumi, N. Machida, J. Chong, M. Fukayama, T. Kodama, and H. Aburatani, "Analyzing Gene Expression Data in Terms of Gene Sets: Methodological Issues," *Bioinformatics,* vol. 23, no. 8, pp. 980-987, 2007.

[20] M. Holec, F. Železný, J. Kléma, and J. Tolar, "Integrating Multiple-Platform Expression Data through Gene Set Features," *Proc. Fifth Int'l Symp. Bioinformatics Research and Applications,* pp. 5-17, 2009.

[21] M. Holec, F. Železný, J. Kléma, and J. Tolar, "A Comparative Evaluation of Gene Set Analysis Techniques in Predictive Classsification of Expression Samples," *Proc. Int'l Conf. Bioinformatics, Computational Biology, Genomics and Chemoinformatics (BCBGC '10),* 2010.

[22] F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J.P. Vert, "Classification of Microarray Data Using Gene Networks," *BMC Bioinformatics,* vol. 8, no. 35, 2007.

[23] E. Lee, H. Chuang, J. Kim, T. Ideker, and D. Lee, "Inferring Pathway Activity Toward Precise Disease Classification," *PLoS Computational Biology,* vol. 4, no. e1000217, 2008.

[24] S. Efroni, C.F. Schaefer, and K.H. Buetow, "Identification of Key Processes Underlying Cancer Phenotypes Using Biologic Pathway Analysis," *PLoS ONE,* vol. 2, no. e425, 2007.

[25] B. Hanczar, M. Courtine, A. Benis, C. Hennegar, K. Clément, and J.-D. Zucker, "Improving Classification of Microarray Data Using Prototype-Based Feature Selection," *SIGKDD Explorations Newsletter,* vol. 5, no. 2, pp. 23-30, 2003.

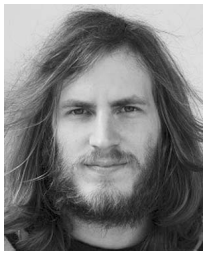
[26] A.L. Tarca, S. Draghici, P. Khatri, S.S. Hassan, P. Mittal, J.-s. Kim, C.J. Kim, J.P. Kusanovic, and R. Romero, "A Novel Signaling Pathway Impact Analysis," *Bioinformatics,* vol. 25, no. 1, pp. 75-82, 2009.

[27] J.P.A. Ioannidis, "Genetic Associations: False or True?," *Trends in Molecular Medicine,* vol. 9, no. 4, pp. 135-138, 2003.

[28] J.P.A. Ioannidis, "Why Most Published Research Findings are False," *PLoS Medicine,* vol. 2, no. e124, 2005.

[29] S.Y. Rhee, V. Wood, K. Dolinski, and S. Draghici, "Use and Misuse of the Gene Ontology Annotations," *Nature Reviews Genetics,* vol. 9, no. 7, pp. 509-515, 2008.

[30] R. Gentleman et al., "Bioconductor: Open Software Development for Computational Biology and Bioinformatics," *Genome Biology,* vol. 5, no. R80, 2004.
[31] J. Cohen, "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement,* vol. 20, pp. 37-46, 1960.
[32] R Development Core Team, *R: A Language and Environment for Statistical Computing,* R Foundation for Statistical Computing, Vienna, Austria, 2010.
[33] J. MacQueen et al., "Some Methods for Classification and Analysis of Multivariate Observations," *Proc. Fifth Berkeley Symp. Math. Statistics and Probability,* vol. 1, no. 14, pp. 281-297, 1967.
[34] A. Thalamuthu, I. Mukhopadhyay, X. Zheng, and G.C. Tseng, "Evaluation and Comparison of Gene Clustering Methods in Microarray Analysis," *Bioinformatics,* vol. 22, no. 19, pp. 2405-2412, 2006.
[35] G. Kerr, H. Ruskin, M. Crane, and P. Doolan, "Techniques for Clustering Gene Expression Data," *Computers in Biology and Medicine,* vol. 38, no. 3, pp. 283-293, 2008.
[36] I. Priness, O. Maimon, and I. Ben-Gal, "Evaluation of Gene-Expression Clustering via Mutual Information Distance Measure," *BMC Bioinformatics,* vol. 8, no. 111, 2007.
[37] F. De Smet, J. Mathys, K. Marchal, G. Thijs, B. De Moor, and Y. Moreau, "Adaptive Quality-Based Clustering of Gene Expression Profiles," *Bioinformatics,* vol. 18, no. 5, pp. 735-746, 2002.
[38] L. Kaufman and P. Rousseeuw, *Finding Groups in Data an Introduction to Cluster Analysis.* Wiley Interscience, 1990.
[39] E. Jones et al., "SciPy: Open Source Scientific Tools for Python," http://www.scipy.org/, 2001.
[40] D. Stirewalt et al., "Identification of Genes with Abnormal Expression Changes in Acute Myeloid Leukemia," *Genes, Chromosomes and Cancer,* vol. 47, no. 1, pp. 8-20, 2008.
[41] A. Tripathi et al., "Gene Expression Abnormalities in Histologically Normal Breast Epithelium of Breast Cancer Patients," *Int'l J. Cancer,* vol. 122, no. 7, pp. 1557-1566, 2008.
[42] Y. Hippo, H. Taniguchi, S. Tsutsumi, N. Machida, J. Chong, M. Fukayama, T. Kodama, and H. Aburatani, "Global Gene Expression Analysis of Gastric Cancer by Oligonucleotide Microarrays," *Cancer Research,* vol. 62, no. 1, pp. 233-240, 2002.
[43] W. Freije, F. Castro-Vargas, Z. Fang, S. Horvath, T. Cloughesy, L. Liau, P. Mischel, and S. Nelson, "Gene Expression Profiling of Gliomas Strongly Predicts Survival," *Cancer Research,* vol. 64, no. 18, pp. 6503-6510, 2004.
[44] T. Bull, C. Coldren, M. Moore, S. Sotto-Santiago, D. Pham, S. Nana-Sinkam, N. Voelkel, and M. Geraci, "Gene Microarray Analysis of Peripheral Blood Cells in Pulmonary Arterial Hypertension," *Am. J. Respiratory and Critical Care Medicine,* vol. 170, no. 8, pp. 911-919, 2004.
[45] R. Palmer et al., "Pediatric Malignant Germ Cell Tumors Show Characteristic Transcriptome Profiles," *Cancer Research,* vol. 68, no. 11, pp. 4239-4247, 2008.
[46] C. Best et al., "Molecular Alterations in Primary Prostate Cancer After Androgen Ablation Therapy," *Clinical Cancer Research,* vol. 11, no. 19, pp. 6823-6834, 2005.
[47] K. Detwiller, N. Fernando, N. Segal, S. Ryeom, P. D'Amore, and S. Yoon, "Analysis of Hypoxia-Related Gene Expression in Sarcomas and Effect of Hypoxia on rna Interference of Vascular Endothelial Cell Growth Factor a," *Cancer Research,* vol. 65, no. 13, pp. 5881-5889, 2005.
[48] B.J. Carolan, A. Heguy, B.-G. Harvey, P.L. Leopold, B. Ferris, and R.G. Crystal, "Up-Regulation of Expression of the Ubiquitin Carboxyl-Terminal Hydrolase l1 Gene in Human Airway Epithelium of Cigarette Smokers," *Cancer Research,* vol. 66, no. 22, pp. 10729-10740, 2006.
[49] B. Bolstad, R. Irizarry, M. Åstrand, and T. Speed, "A Comparison of Normalization Methods for High Density Oligonucleotide Array Data Based on Variance and Bias," *Bioinformatics,* vol. 19, no. 2, pp. 185-193, 2003.
[50] T. Barrett, D. Troup, S. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. Kim, A. Soboleva, M. Tomashevsky, and R. Edgar, "Ncbi Geo: Mining Tens of Millions of Expression Profiles-Database and Tools Update," *Nucleic Acids Research,* vol. 35, no. suppl 1, pp. D760-D765, 2007.
[51] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," *Proc. Int'l Joint Conf. Artificial Intelligence,* pp. 1137-1143, 1995.
[52] V. Vapnik, *The Nature of Statistical Learning Theory.* Springer Verlag, 2000.
[53] L. Breiman, "Random Forests," *Machine Learning,* vol. 45, no. 1, pp. 5-32, 2001.
[54] J. Quinlan, *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.
[55] I. Rish, "An Empirical Study of the Naive Bayes Classifier," *Proc. IJCAI Workshop Empirical Methods in Artificial Intelligence,* pp. 41-46, 2001.
[56] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," *IEEE Trans. Information Theory,* vol. 13, no. 1, pp. 21-27, 1967.
[57] M. Brown, W. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. Furey, M. Ares, and D. Haussler, "Knowledge-Based Analysis of Microarray Gene Expression Data by using Support Vector Machines," *Proc. Nat'l Academy of Sciences USA,* vol. 97, no. 1, pp. 262-267, 2000.
[58] R. Díaz-Uriarte and S. De Andres, "Gene Selection and Classification of Microarray Data Using Random Forest," *BMC Bioinformatics,* vol. 7, no. 3, 2006.
[59] J. Demšar, B. Zupan, G. Leban, and T. Curk, "Orange: From Experimental Machine Learning to Interactive Data Mining," *Proc. Conf. Principles and Practice of Knowledge Discovery in Databases (PKDD '04),* pp. 537-539, 2004.
[60] F. Wilcoxon, "Individual Comparisons by Ranking Methods," *Biometrics,* vol. 1, no. 6, pp. 80-83, 1945.
[61] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *J. Machine Learning Research,* vol. 7, pp. 1-30, 2006.
[62] F.D. Gibbons and F.P. Roth, "Judging the Quality of Gene Expression-Based Clustering Methods Using Gene Annotation," *Genome Research,* vol. 12, no. 10, pp. 1574-1581, 2002.
[63] M. Friedman, "The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance," *J. Am. Statistical Assoc.,* vol. 32, no. 200, pp. 675-701, 1937.
[64] Y. Saeys, I.n. Inza, and P. Larrañaga, "A Review of Feature Selection Techniques in Bioinformatics," *Bioinformatics,* vol. 23, no. 19, pp. 2507-2517, 2007.
[65] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering Aggregation," *Proc. 21st Int'l Conf. Data Eng.,* pp. 341-352, 2005.
[66] A. Strehl and J. Ghosh, "Cluster Ensembles—A Knowledge Reuse Framework for Combining Multiple Partitions," *The J. Machine Learning Research,* vol. 3, pp. 583-617, 2003.
[67] P. Glenisson, J. Mathys, and B. de Moor, "Meta-clustering of Gene Expression Data and Literature-Based Information," *SIGKDD Explorations,* vol. 5, pp. 101-112, 2003.
[68] J. Tomfohr, J. Lu, and T.B. Kepler, "Pathway Level Analysis of Gene Expression Using Singular Value Decomposition," *BMC Bioinformatics,* vol. 6, no. 225, 2005.



Miloš Krejník received the BSc degree in computer technology and the MSc degree with honors in cybernetics and measurement from the Czech Technical University in Prague (CTU), in 2006 and 2008, respectively. Currently, he is working toward the PhD degree in artificial intelligence and biocybernetics at CTU. In 2007, he was a researcher in the Gerstner Laboratory at CTU. From 2009 to 2011, he was a quantitative analyst at Analytical Department at RSJ, Prague, Czech Republic. His research interests focus on statistical machine learning in bioinformatics and finance.

Jiří Kléma received the PhD degree in artificial intelligence and biocybernetics from the Czech Technical University in Prague (CTU) in 2002. From 2005 to 2006, he carried out postdoctoral training at the University of Caen, France. Currently, he is an assistant professor at CTU. His main research interest is data mining and its applications in industry, medicine, and bioinformatics. He focuses namely on knowledge discovery and learning in domains with heterogeneous and complex background knowledge. He is a coauthor of 15 journal publications and book chapters, a reviewer for several international journals and a member of the Presidium of The Czech Society for Cybernetics and Informatics.

Chapter 4

Information Extraction from Genomic Texts

Automated Information Extraction from Gene Summaries

Thierry Charnois, Nicolas Durand, and Jiří Kléma

GREYC, CNRS - UMR 6072, Université de Caen
Campus Côte de Nacre, F-14032 Caen Cédex France
{Forename.Surname}@info.unicaen.fr

Abstract. Automated extraction of links among biological entities from free biological texts has proven to be a difficult task. In this paper we propose and solve a modified task in which we extract the links from short textual gene summaries collected automatically from NCBI website. The main simplification lies in the fact that each summary is unambiguously attached to a single gene. The agent part of binary biological interactions is thus known by default, the goal is to identify meaningful target parts from the summary. The outcome is a structured representation of each summary that can be used as background knowledge in consequent mining of gene expression data. As the gene summaries highly interact with the other structural information resources provided by NCBI website, these resources can be used as an annotation tool and/or a feedback for performance optimization of the system being developed. In particular we use the gene ontology terms in order to evaluate and improve the information extraction process.

Keywords: genomics, text mining, biological information extraction.

1 Introduction

As availability of textual information related to biology increases, research on information extraction (IE) is rapidly becoming an essential component of various bio-applications. It is expected that text mining in general, and IE in particular, will provide tools to facilitate the annotation of a large amount of genetic information, including gene sequences, transcription profiles and biological pathways. The biological function of cells, tissues and organisms can be understood by examination of interactions among proteins or between DNA and proteins.

The main interest has been devoted to MedLine abstracts, however there is also a vast effort to exploit full-text journal articles [20]. Applying IE to genomics and more generally to biology is not an easy task because IE systems require deep analysis methods to extract the relevant pieces of information. That is why we propose a modified task in which we extract the links from short textual gene summaries collected automatically from NCBI website. The main simplification lies in the fact that each summary is unambiguously attached to a single gene. The agent part of binary biological interactions is thus known by default, the goal is to identify meaningful target parts from the summary.

This work has started with the intention to develop a meaningful measure of interaction inside a closed set of genes in order to support consequent mining of gene expression data. Such a measure can be used in many ways. The measure can complement the gene distance measure based immediately on the expression data when the genes are clustered [15]. It can be used to select biologically meaningful patterns from the overwhelming pattern sets that technically appear in the expression data [17] or it can help in feature extraction and selection when a classification task is solved [24].

Public databases contain vast amount of rich data that can be used to create and evaluate both direct and indirect interactions among biological entities. Of course, the most straightforward way is to utilize the structured information such as gene ontology (GO) or Entrez's link files. The rationale sustaining the GO based measure is that the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related. [19] defines a distance based on the Czekanowski-Dice formula, the methodology is implemented within the GOProxy tool of GOToolBox [1]. [22] uses Entrez's link files in order to create a general entity graph. The authors also provide a measure that assesses the strength of a link between an arbitrary pair of vertices.

Nevertheless, the structured databases can hardly summarize all the available knowledge and text mining outcomes can reasonably complement the information gained from the knowledge sources mentioned in the previous paragraph. Tagging gene and protein names in free biomedical text has proven to be a difficult task [23]. Automated extraction of direct links among biological entities is even more difficult [10]. In this paper we restrict to a corpus of gene textual summaries. Possible interaction among a closed set of genes is studied indirectly. The main aim of the paper is to develop a structured and tagged representation of gene summaries. This structured representation can later serve to assume on interaction or similarity among the genes from multiple points of view. The proposed structured representation also seems to be promising with respect to its further generalization. The developed system provides an insight into relations among biological entities and it can be adjusted to extract arbitrary interactions among biological entities from abstracts or whole texts.

This paper is structured as follows. Section 2 briefly introduces the data we worked with. Section 3 gives an overview of related methods. Section 4 describes a tool LinguaStream that we have used, discusses the developed extraction rules and gives examples of real outputs. Section 5 provides two ways of evaluation – the first is based on a limited corpus of human annotated summaries, the second evaluates the full corpus with respect to GO terms.

2 Entrez Gene Summaries

Entrez Gene is the gene-specific database at the National Center for Biotechnology Information (NCBI), a division of the National Library of Medicine (NLM). Entrez Gene provides unique integer identifiers for genes and other loci for a subset of model organisms. It tracks those identifiers, and is integrated with the Entrez system for interactive query, LinkOuts, and access by E-utilities [25].

The information that is maintained includes nomenclature, chromosomal localization, gene products and their attributes (e.g. protein interactions), associated markers, phenotypes, interactions, and a wealth of links to citations, sequences, variation details, maps, expression reports, homologs, protein domain content and external databases.

As mentioned in Section 1 the long term goal is to develop a meaningful measure of interaction inside a closed set of genes. In our experiments we deal with the SAGE human gene expression dataset downloaded from [4]. Only the unambiguous tags (corresponding to genes) identified with RefSeq were selected, leaving a set of 11082 tags (expressed in 207 biological situations).

To access the gene annotation data for every tag considered, RefSeq identifiers were translated into EntrezGene identifiers [3]. The mapping approached 1 to 1 relationship. There were only 11 unidentified RefSeqs, 24 RefSeqs mapped to more than 1 id and 203 ids still appeared more than once. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the Entrez Gene database [4] and sequentially parsed by the method stemming from [28]. The non-trivial textual records were obtained for 6,302 ids which makes 58% of the total amount of 10,858 unique ids. 3,926 genes had a short summary, 5,109 had one abstract attached at least. 6,824 genes had at least a single GO term attached, which makes 63% of the total amount of genes.

3 Information Extraction and Methodology

Many approaches have been proposed for extraction of biological information from scientific texts. These approaches can be classified into two broad categories [8]: machine learning based and linguistic analysis based. That is the latter one, and more precisely IE technique, which is used in this paper. Some of the IE systems use similar approaches with the Natural Language Processing (NLP) understanding systems of the seventies/eighties and IE is often seen as a NLP understanding system. In fact, they do not share the same goal. IE aims at extracting very precise information from a restricted domain while the goal of the NLP systems was the whole understanding of all aspects of the text. For this purpose, extensive knowledge and linguistic resources were needed, and deep analysis was necessary (syntactic, semantic and pragmatic analysis).

Taking advantage of the restricted domain, some biomedical IE systems adopt this NLP based architecture [11, 14]. Nevertheless, the syntactic analysis still remains a difficult task. Actually the accuracy of the complete parsing can be estimated roughly about 50% of the analysed sentences (see [11]). Other works attempt to use "shallow parsing", a robust method, although less precise performing a partial decomposition of a sentence structure to identify phrasal chunks or entities of interest and relations between these entities. Generally, these kinds of systems are designed for extracting protein-protein relations, such as protein-location relations, binding relations, gene-gene interactions, etc. [21]. A common point of the IE systems is that they utilize resources, biological databases, ontologies, such as UMLS, LocusLink ...

Some other papers are devoted to a preliminary task: the recognition of gene/protein names and families. Difficulties are well known: multi-sense words, no formal criterion, multi-word terms, variations in gene/protein names. Different NLP methods are used for this like rule-based approach [12], or/and dictionary/knowledge approach [16, 18].

Our system differs from the approaches previously mentioned in several ways. For example, [14] carries out a terminological parsing, using a biological knowledge database, syntactic, semantic and discoursive analysis, using a domain model (ontology) to get a predicate argument representation and to fill an extraction template. Instead of those kinds of classical NLP techniques, we design simple declarative extraction rules, making the implementation process "light and quick". Let us note that they are domain-specific, but by no means corpus-specific. One of the aims is to reach similar results as the "heavy" methods published in the literature.

The design of the rules can be seen as a simplification of the "contextual exploration method" [26]. This approach aims at locating contexts in a corpus (i.e., linguistic indicators) from which some rules for identifying relevant textual segments are triggered. For example, linguistic indicators as "our conclusion", "consequently" or "so" found in a corpus can allow the extraction of conclusive sentences. That is the idea of our system: triggering extraction rules only if a context is located while avoiding the whole-corpus analysis. Another similarity with our work is that no syntactic analysis is processed. However, unlike our approach, this method is domain-independent, so linguistic indicators and extracted informations are general (causality, thematic announcement, conclusive sentences, ...).

Another important point is that our method is endogenous: no resources such as knowledge base or dictionary are needed at the beginning. The resources are constructed on the fly – the system learns new terms (which can be new terms in the domain or missing in the databases) to be used later or in other biological corpora and/or in other text mining applications.

Finally, our system is not designed to focus only on a specific aspect of gene/protein description but it is designed to identify protein/family/name and general biological function about the gene/protein involved. Actually four *types* of information are distinguished and annotated in the corpus: gene/protein name, family name, location and biological function.

4 Method

The presented approach consists in definition of extraction rules, and has been implemented using the LinguaStream platform.

4.1 LinguaStream

LinguaStream [2, 7] is an integrated experimental environment targeted to NLP researchers. It allows complex experiments on corpora to be realised conveniently, using various declarative formalisms.

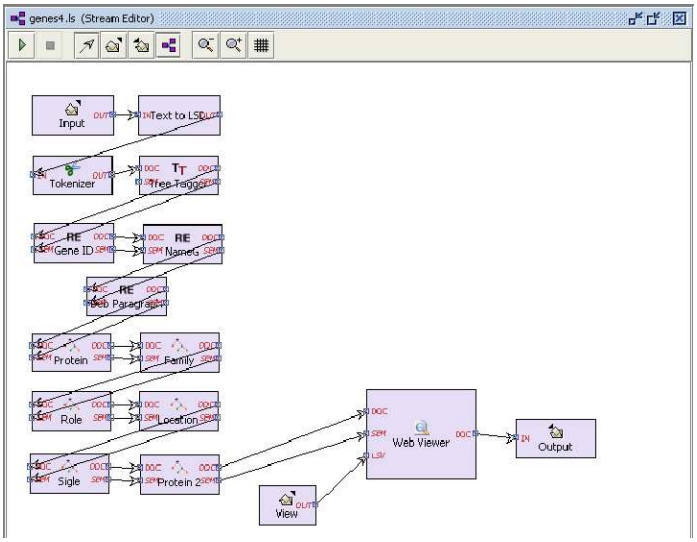


Fig. 1. Processing stream of the implemented rules in LinguaStream.

Its integrated environment allows processing streams to be assembled visually (see Figure 1), picking individual components from a "palette". Some components are specifically targeted to NLP, while others solve various issues related to document engineering (especially to XML processing). Annotations made on a single document are organized in independent layers and may overlap. Thus, concurrent and ambiguous annotations may be represented in order to be solved afterwards, by subsequent analysers. The platform is systematically based on XML recommendations and tools, and is able to process any file in this format while preserving its original structure. When running a processing stream, the platform takes care of the scheduling of sub-tasks, and various tools allow the results to be visualised conveniently.

IE from a raw text is composed of tokenization, POS tagging (using TreeTagger [5]), extraction and output generation which adds the final XML wrapper. Among fundamental principles, the platform allows the **declarative representations** to be used. Furthermore, the **modularity** of processing streams promotes the **reusability** of components in various contexts: a given module, developed for a first processing stream may be used in other ones. Section 4.2 demonstrates their utility.

4.2 Extraction rules

We have defined a set of rules to identify, extract and annotate relevant multiword terms from gene summaries. The results are given in a form of XML file containing the whole text where the recognized areas are highlighted and clickable (see Figure 2), and another XML file with the extracted information only (see Figure 3). Let us refer to these files as the *interactive* and *extracted* output.

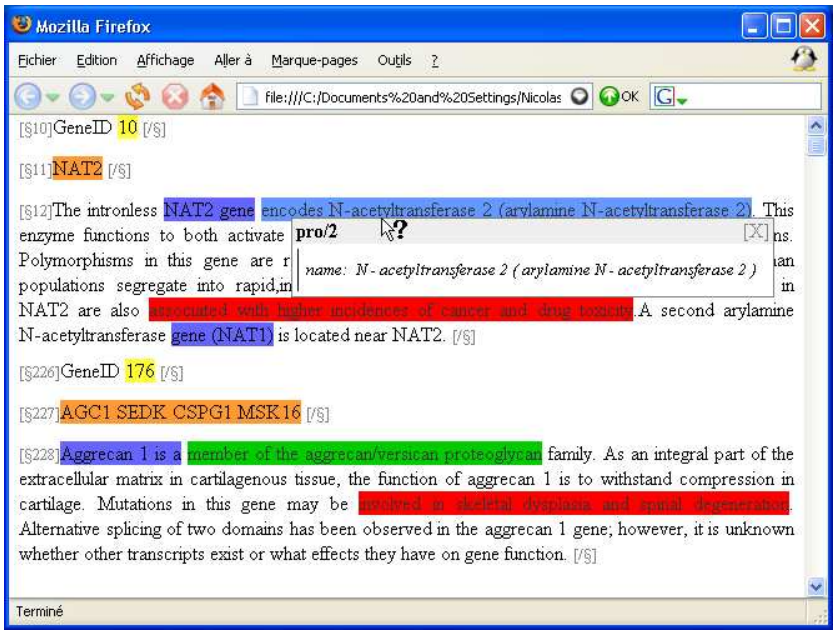


Fig. 2. Example of XML result.

The rule definition is decomposed into the following steps:

- Observe the corpus (in fact the training corpus) in order to get regularities and identify some contexts. For example, the expression "this gene encodes the X protein" has numerous occurrences in the corpus, so "encode" is a good context – a trigger word – to identify a protein name;
- Design rules from the contexts previously identified (a particular example is shown later).
- Implement the rules. It is a straightforward process using DCG Prolog and unification on feature structures thanks to GULP [9].
- Review the results after processing the rules and backtrack if necessary. This can be changing rules, or adding rules while possibly reusing the terms/knowledge already recognized/learned.

We have defined 4 sets of rules allowing the system to recognize 4 types of information: protein names, family names of proteins, roles / biological functions (including diseases, interactions, ...), and location (components, ...).

General structure of the rules We do not use patterns in the sense of the IE, that is without an a priori on the form of the expressions. Figure 4 presents the structure of the rules. From a "context", an expression (generally a multi-word term, a nominal phrase) is recognized until a stop phrase is encountered. The context is a set of "trigger" words. The stop phrases can be words, symbols, verbs, punctuation, ... They depend on the rule type.

```
<gene>
<id>10</id>
<name>NAT2</name>
<protein>encodes N-acetyltransferase 2 (arylamine N-acetyltransferase 2)</protein>
<role>associated with higher incidences of cancer and drug toxicity</role>
</gene>
<gene>
<id>176</id>
<name>AGC1 SEDK CSPG1 MSK16</name>
<protein>Aggrecan 1 is a</protein>
<family>member of the aggrecan/versican proteoglycan</family>
<role>involved in skeletal dysplasia and spinal degeneration</role>
</gene>
```

Fig. 3. The extracted results.

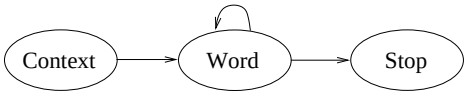


Fig. 4. Structure of the rules.

Let us take an example. The term "encodes N-acetyltransferase 2 (arylamine N-acetyltransferase 2)." is extracted by using the following set of rules:

```
protein(type:pro..name:N) --> @lemma:encode, np(N).
np(N) --> @tag:dt, namepro(N).
np(N) --> namepro(N).
namepro(N) --> ls_token(N,_), end.
namepro(N) --> ls_token(N,_), namepro(N2), {concat(N1,N2,N)}.
end --> punctuation ; verb ; relative_pronoun ; trigger_word.
```

In this example, "namepro" stands for the name of the protein to be extracted, "ls_token" a terminal symbol (a token), "end" the indicator for cutting out the recognition of a multi-word term. The trigger phrase is "encode" (i.e., also encoded, encodes atc.). Let us remark that "dt" corresponds to a possible determiner just before the name of the protein. The rules "namepro" allow the system to recognize the multi-word terms. The end phrase is a punctuation symbol here.

Context and stop phrases Currently, the identification of the context has been done manually, however an automatic learning of the context can also be considered. We have manually detected special phrases for each type of information (proteins, families, ...) on an excerpt of the corpus. We have noted the corresponding trigger words and the stop phrases. Table 1 presents some examples of trigger phrases for each type. The common stop phrases are the trigger words, some punctuation symbols and the relative pronouns.

The final rulebase consists of 186 rules – 74 for proteins, 46 for families, 27 for roles, 39 for locations.

Special processing Another set of rules benefits from the reusability principle of LinguaStream. It enables us to use the information (tokens, trigger

phrases, ...) recognized earlier within the current processing stream. Some protein names/families are recognized using entities already identified. These entities are considered as lexical units (tokens) in the rules. For instance, "`X are class of FAMILY`" where `FAMILY` is the entity previously recognized as a protein family, and `X` is the new extracted information: here an other protein family.

A special process for recognizing protein names expressed by an acronym is done. All acronyms are marked by a few special rules that extract words with upper cases, numerical figures and/or special symbols as in [12]. Then, the acronym context is used to decide whether the acronym corresponds to a protein name. For example, "protein CEBP-alpha" is detected using these specific rules. Here the rule is: the word "protein" followed by an acronym. We also use particular rules to filter out false or misleading expressions. For instance, the term "the secreted protein" must not be extracted as a protein name.

proteins	encodes an ...	@lemma:encode, @tag:dt
	the product of this gene is ...	@lemma:product, $'of', $'this', $'gene', @lemma:be
families	belongs to the ...	@lemma:belong, $'to'
	is a member of the ...	@lemma:member, $'of'
roles	an important role in ...	$'role', $'in'
	is involved in ...	$'involved', $'in'
locations	found in ...	$'found', $'in'
	located in ...	$'located', $'in'

Table 1. Examples of detected contexts.

4.3 Outcome

The corpus is about 2.33MB and contains 64,308 lines. There are 10,858 genes. In order to learn the contexts and the stop phrases, we have looked over 200 genes (1.8% of the corpus).

Type	No. text areas
proteins	3,058
families	3,056
roles	4,303
locations	1,023

Table 2. The number of recognized terms (text areas) according to their type.

The number of marked text areas (i.e., multi-word terms or pieces of extracted information) is presented in Table 2. Let us note that in a gene summary, the information about proteins, families, ...is not always present. We observe that the number of "roles" is larger than the number of "proteins". As a matter of fact, a single gene may have several "roles" because the type role contains biological functions, diseases and also different interactions. As regards families, we capture families and also subfamilies and superfamilies, if they are indicated.

The system has recognized 3,058 protein names. By rule of thumb, this is a relatively good result since there are 6,932 genes without summaries (i.e. without any chance to extract information). On average, there is nearly one protein name

extracted per existing summary. A more detailed evaluation of the performance is given in the next Section.

5 Evaluation

Two types of experiments have been carried out. First, we have evaluated the precision and the recall of the method using an excerpt of the data. The second experiment is a direct comparison between our extracted terms and the GO terms annotating the individual genes.

5.1 Evaluation on a human annotated corpus

We have evaluated our approach using 100 genes (and the corresponding summaries) randomly chosen. This excerpt has been annotated by two local experts to form the reference. We have computed the classical measures of precision and recall [8] to assess the performance of our system.

Table 3 presents the results and relates them to the results obtained by other existing methods published in literature. The comparison is illustrative only as the methods were not applied to the same corpus. The precision and recall values cannot be compared directly, but they may give an estimation of the performance. As we can see, the results of our system are comparable to the existing scores, without using a "heavy method" nor resources.

method		recall	precision
existing methods: [12, 18, 13, 27]		73-99%	73-95%
our approach:	proteins	73,6%	78,8%
	families	71,6%	93,4%

Table 3. Results.

Distinguishing various term types, a good precision and recall has been reached for families. Actually, for the family names and the locations, the implemented rules are appropriate to extract information from summaries. Moreover, we have recently improved the rules by using the results of this evaluation and by observing the information not recognized to define new contexts.

As for biological functions, the important point is to have a relatively complete list of the commonly used verbs. Our "list" is good enough, and it is easy to add new verbs to capture more cases. For this, ideas from specific works like [6] can be used.

The main problem concerns the recognition of proteins names. The current rules are able to capture the majority of the names, however some particular linguistic problems are not treated yet: anaphoras and coordination. For instance, in the phrase "the related proteins CEBP-alpha, CEBP-delta, and CEBP-gamma" all the three acronyms are recognized but only the first one ("CEBP-alpha") is identified as a protein name (the rule given above). The others would need to take the coordination problem into account.

5.2 Comparison with GO

A great part of GO terms associated with genes also appear in their summaries. In other words, if a gene is annotated with a GO term, this term (or its semantically equivalent phrase) often appears in the summary of the given gene too. This significant overlap between GO terms and summaries gives us a chance to utilize GO terms as an annotation tool for gene summaries. The quality of information extraction can be tested with respect to recall of the known GO terms. The main advantage of such an experiment is that it enables us to automatically evaluate the system over whole the corpus of gene summaries.

The basic assumption of this evaluation is that all the GO terms represent meaningful terms to be extracted. Then, the recall is estimated as the ratio between the number of GO terms identified within the extracted XML annotation and the number of GO terms that appear within the original summaries. The ideal case occurs when all the GO terms that appear within the gene summary of the given gene remain also in its XML record – recall would be 1 here.

The main and difficult problem is to identify the GO terms within free text of gene summaries. First, let us see what is the percentage of GO terms that appear in gene summaries immediately – as exactly the same term or phrase. The simple search for substrings suggests that only 7% of GO terms associated with the given gene co-appear in its summary immediately. These are mainly one word terms since for longer phrases the exact match is less likely – e.g., the GO term "amino acid metabolism" appears in the summary as an expression "function in the catabolism and salvage of acylated amino acids". That is why we have also applied a simple form of approximate match for longer phrases. If at least one of the stemmed words from the GO phrase appears in the gene summary exactly, we search for an approximate match of the other words in the same summary sentence. We use the bigram approximate string comparison for this purpose. The phrase is found if and only if the average of best-match values – we search for the nearest counterpart for all the words from the GO phrase – reaches a certain threshold. This simple approximate match reveals that 18% of GO terms associated with the given gene co-appear in the respective summary.

matching	original summaries	extracted XML	recall
exact	7%	3.9%	56%
approximate	18%	8.2%	46%

Table 4. Recall of GO terms – the exact and approximate match.

Table 4 gives an overview of the recall for exact and approximate GO terms. Precision of the IE cannot be revealed in this way as we do not search for GO terms only. Let us remind that we are interested in any biological terms and the goal is not to confine ourselves to the limited dictionary of GO terms. Nevertheless, the recall presented in Table 4 should be evaluated with respect to the condensation that the extraction process brings. The content of the extracted output makes 29.2% of the content of the original gene summary files. It also has to be considered that the sentences in gene summaries can be quite long while the extracted tags are quite compact. The chance that the GO phrase is scattered by tag tokens is thus increased.

6 Conclusion

In this paper, we presented an original approach for extracting and exploiting information from biological domain. The approach gives promising results on a specific but wide corpus. It is suitable for extracting biological information as well as to acquire knowledge. It also seems to be promising with respect to its further generalization. The developed grammar provides an insight into relations among biological entities and it can be adjusted to extract arbitrary interactions among biological entities from abstracts or whole texts using the acquired information such as terminological resources. The second investigation will consist in enhancing the grammar by an automated learning of context [6]. The process has not been designed yet but the terms already learned can guide and accelerate the learning process (to annotate the other corpus, etc.).

Another work is to propose and test a gene similarity measure based on the developed structured representation. While the measure itself is more or less obvious – the more overlap two genes show in their corresponding type fields the more they interact – the main effort will be to show a possible difference in comparison with the measures that are immediately based on the GO annotations [19] or a vector representation of whole summaries [17].

Acknowledgements. The authors thank A. Widlöcher and F. Bilhaut (the Linguastream team), and the CGMC Laboratory (CNRS UMR 5534, Lyon, France) for providing the gene expression database. This work has been partially funded by the ACI "masse de données" (French Ministry of research), Bingo project (MD 46, 2004-2007).

References

[1] GOToolBox website: http://crfb.univ-mrs.fr/gotoolbox/.
[2] LinguaStream website: http://www.linguastream.org/.
[3] MatchMiner website: http://discover.nci.nih.gov/matchminer/.
[4] NCBI website: http://www.ncbi.nlm.nih.gov/.
[5] TreeTagger website: http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.
[6] P. Bessières, G. Bisson, A. Nazarenko, C. Nédellec, M. Ould Abdel Vetah, and T. Poibeau. Ontology Learning for Information Extraction in Genomics Bibliography - the Caderige Project. In *Journées IMPG Ontologie et Extraction d'Information en Génomique*, Grenoble, France, May 2001.
[7] F. Bilhaut and A. Widlöcher. LinguaStream: An Integrated Environment for Computational Linguistics Experimentation. In *the European Chapter of the Association of Computational Linguistics (Companion Volume)*, Trento, Italy, 2006.
[8] K. B. Cohen and L. Hunter. *Artificial Intelligence Methods and Tools for Systems Biology*, volume 5, chapter Natural Language Processing and Systems Biology. Springer Verlag, 2004.
[9] M. A. Covington. GULP 3.1: An Extension of Prolog for Unification-Based Grammar, 1994.
[10] J. Cussens and C. Nédellec, editors. *Proceedings of the 4th Learning Language in Logic Workshop (LLL05)*, Bonn, August 2005.
[11] N. Daraselia, S. Egorov, A. Yazhuk, S. Novichkova, A. Yuryev, and I. Mazo. Extracting Protein Function Information from MEDLINE Using a Full-Sentence

Parser. In *ECML/PKDD Workshop on Data Mining and Text Mining for Bioinformatics*, Pisa, Italy, Sept. 2004.

[12] K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi. Toward Information Extraction: Identifying Protein Names from Biological Papers. In *Pacific Symposium Biocomputing (PSB'98)*, pages 362–373, Hawaii, Jan. 1998.

[13] K. Fundel, D. Güttler, R. Zimmer, and J. Apostolakis. A Simple Approach for Protein Name Identification: Prospects and Limits. *BMC Bioinformatics*, 6(Suppl 1), 2005.

[14] R. J. Gaizauskas, G. Demetriou, P. J. Artymiuk, and P. Willett. Protein Structures and Information Extraction from Biological Texts: The PASTA System. *Bioinformatics*, 19(1):135–143, 2003.

[15] P. Glenisson, J. Mathys, and B. D. Moor. Meta-Clustering of Gene Expression Data and Literature-Based Information. *SIGKDD Explor. Newsl.*, 5(2):101–112, 2003.

[16] K. Humphreys, G. Demetriou, and R. Gaizauskas. Two Applications of Information Extraction to Biological Science Journal Articles: Enzyme Interactions and Protein Structures. In *Pacific Symposium Biocomputing*, pages 505–516, Hawaii, Jan. 2000.

[17] J. Kléma, A. Soulet, B. Crémilleux, S. Blachon, and O. Gandrillon. Mining Plausible Patterns from Genomic Data. In *the 19th IEEE International Symposium on Computer-Based Medical Systems*, pages 183–188, Salt Lake City, Utah, 2006.

[18] A. Koike and T. Takagi. Gene/Protein/Family Name Recognition in Biomedical Literature. In *Linking Biological Literature, Ontologies and Databases: Tools for Users, Workshop in conjunction with NAACL / HLT 2004*, pages 9–16, 2004.

[19] D. Martin, C. Brun, E. Remy, P. Mouren, D. Thieffry, and B. Jacq. GOToolBox: Functional investigation of gene datasets based on gene ontology. *Genome Biology*, 5(12):R101, 26 Nov. 2004.

[20] S. K. Parantu, P.-I. Carolina, B. Peer, and A. A. Miguel. Information extraction from full text scientific articles: Where are the keywords? *BMC Bioinformatics*, 4:20, 2003.

[21] J. Pustejovsky, J. Castano, J. Zhang, M. Kotecki, and B. Cochran. Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations. In *Pacific Symposium on Biocomputing (PSB'02)*, pages 362–373, Hawaii, Jan. 2002.

[22] P. Sevon, L. Eronen, P. Hintsanen, K. Kulovesi, and H. Toivonen. Link Discovery in Graphs Derived from Biological Databases. In *3rd International Workshop on Data Integration in the Life Sciences 2006 (DILS'06)*, Hinxton, UK, July 2006.

[23] L. Tanabe and W. J. Wilbur. Tagging gene and protein names in biomedical text. *Bioinformatics*, 18:1124–1132, 2002.

[24] J.-P. Vert and M. Kanehisa. Graph-Driven Feature Extraction From Microarray Data Using Diffusion Kernels and Kernel CCA. In S. Becker, S. Thrun, and K. Obermayer, editors, *NIPS*, pages 1425–1432. MIT Press, 2002.

[25] D. Wheeler, D. Benson, and S. Bryant. Database Resources of the National Center for Biotechnology Information: Update. *Nucleic Acids Res.*, 33:D39–D45, 2005.

[26] D. Wonsever and J.-L. Minel. Contextual Rules for Text Analysis. In *CICLing '01: Proceedings of the Second International Conference on Computational Linguistics and Intelligent Text Processing*, pages 509–523, London, UK, 2001. Springer.

[27] H. Yu, V. Hatzivassiloglou, C. Friedman, A. Rzhetsky, and W. Wilbur. Automatic Identifying Gene/Protein Terms in MEDLINE Abstracts. *Journal of Biomedical Informatics*, 35(5-6), 2002.

[28] F. Zelezny, J. Tolar, N. Lavrac, and O. Stepankova. Relational Subgroup Discovery for Gene Expression Data Mining. In *EMBEC: 3rd IFMBE European Medical & Biological Engineering Conf.*, November 2005.

Combining sequence and itemset mining to discover named entities in biomedical texts: a new type of pattern

Marc Plantevit and Thierry Charnois

Université de Caen Basse Normandie,
CNRS, UMR6072, GREYC F-14032, France
Fax: +33231567330
E-mail: marc.plantevit@info.incaen.fr
E-mail: thierry.charnois@info.incaen.fr

Jiří Kléma

Faculty of Electrical Engineering,
Czech Technical University,
Technická 2, Prague 6, 166 27, Czech Republic
E-mail: klema@labe.felk.cvut.cz

Christophe Rigotti

Université de Lyon, CNRS,
INSA-Lyon, LIRIS, UMR5205, F-69621, France
E-mail: christophe.rigotti@insa-lyon.fr

Bruno Crémilleux*

Université de Caen Basse Normandie,
CNRS, UMR6072, GREYC F-14032, France
Fax: +33231567330
E-mail: bruno.cremilleux@info.unicaen.fr
*Corresponding author

Abstract: Biomedical named entity recognition (NER) is a challenging problem. In this paper, we show that mining techniques, such as sequential pattern mining and sequential rule mining, can be useful to tackle this problem but present some limitations. We demonstrate and analyse these limitations and introduce a new kind of pattern called LSR pattern that offers an excellent trade-off between the high precision of sequential rules and the high recall of sequential patterns. We formalise the LSR pattern mining problem first. Then we show how LSR patterns enable us to successfully tackle biomedical NER problems. We report experiments carried out on real datasets that underline the relevance of our proposition.

Keywords: LSR patterns; sequential patterns; biomedical named entity recognition problem; NER; constraint-based pattern mining.

Reference to this paper should be made as follows: Plantevit, M., Charnois, T., Kléma, J., Rigotti, C. and Crémilleux, B. (2009) 'Combining sequence and itemset mining to discover named entities in biomedical texts: a new type of pattern', *Int. J. Data Mining, Modelling and Management*, Vol. 1, No. 2, pp.119–148.

Biographical notes: Marc Plantevit is a Postdoctoral Researcher in Computer Science at GREYC Laboratory (CNRS UMR 6072), University of Caen, France. He is currently working on the ANR (French National Research Agency) funded project Bingo2 ANR-07-MDCO-014. He received his PhD in Computer Science in 2008 from the University of Montpellier, France, in the field of sequential pattern mining. His main research interests include sequential pattern mining, text mining and knowledge discovery in multidimensional databases.

Thierry Charnois is an Assistant Professor at IUT Caen and at the GREYC Laboratory (CNRS UMR 6072), University of Caen, France. He holds a PhD in Computer Science (1999) from LIPN Laboratory, University of Paris 13, in the field of natural language processing (NLP). His research interests include NLP and semantics, discovery knowledge and information extraction from texts, and its applications to the biomedical domain.

Jiří Kléma received his PhD in Artificial Intelligence and Biocybernetics from the Czech Technical University (CTU) in Prague in 2002. In 2005–2006, he carried out Post-Doctoral training at the University of Caen, France. Currently, he is an Assistant Professor at CTU. His main research interest is data mining and its applications in industry, medicine and bioinformatics. He focuses namely on knowledge discovery and learning in domains with heterogeneous and complex background knowledge. He is a co-author of 15 journal publications and book chapters, a Reviewer for several international journals and a member of the Presidium of The Czech Society for Cybernetics and Informatics.

Christophe Rigotti is an Assistant Professor at INSA-Lyon (University of Lyon) and works at the LIRIS laboratory (UMR5205 CNRS). He holds a PhD in Computer Science (1996) from INSA-Lyon/University of Lyon, in the field of object-oriented and deductive databases. Since then, he has been working on multi-dimensional databases, constraint programming and data mining. In data mining, his main research interests include sequential pattern mining and condensed representations for pattern extraction. He is a co-author of over 40 papers in journal and conference proceedings and he is member of many international committees.

Bruno Crémilleux is a Professor in Computer Science at GREYC Laboratory (CNRS UMR 6072), University of Caen, France. He is currently heading the data mining research group. He received his PhD in 1991 from the University of Grenoble, France and a Habilitation degree in 2004 from the University of Caen. His current research interests include knowledge discovery in databases, machine learning, text mining and their applications notably in bioinformatics and medicine. He is a co-author of over 50 papers in journals and conference proceedings and he is a member of many international conference committees.

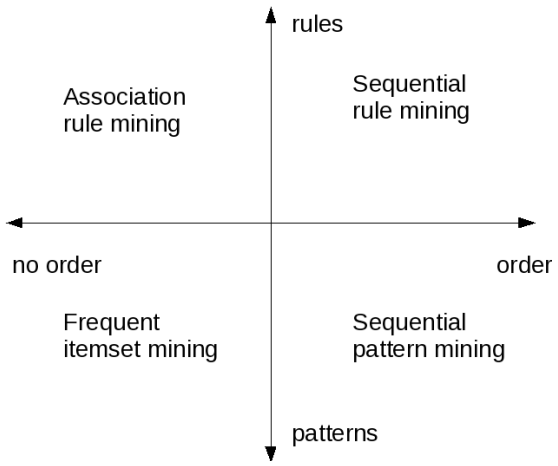
1 Introduction

In current scientific, industrial or business areas, one of the critical needs is to derive knowledge from huge datasets or text collections. This task is at the core of the knowledge discovery in database (KDD) area. In particular, a large part of the biological information is available in natural language in research publications, technical reports, websites and other text documents. A critical challenge is then to extract relevant and useful knowledge dispersed in such text collections. Lots of efforts have been made such as designing efficient tools to tackle large datasets and the discovery of patterns (i.e., relationships in the data) of potential interest to the user. Text mining in general and information extraction (IE) in particular are rapidly becoming an essential component of various bio-applications. These techniques and natural language processing (NLP) have been widely applied to extract and exploit background knowledge from biomedical texts. Among many tasks, a crucial issue is the annotation of a large amount of genetic information. IE and NLP aim at processing accurate parsing to extract specific knowledge such as named entities (e.g., gene, protein) and relationships between the recognised entities (e.g., gene-gene interactions, biological functions). The need of linguistic resources (biological databases, ontologies and IE rules such as grammars or patterns) is a common feature of the methods provided by the literature. Difficulties are well-known: multi-sense words, no formal criterion, multi-word terms and variations in gene/protein names. These linguistic issues are often handled using rules. But, except very few attempts (Califf and Mooney, 1999; Smith et al., 2008), such rules are manually elaborated and texts, which can be processed are necessarily specific and limited. Furthermore, machine learning (ML) based methods such as support vector machines, conditional random fields, etc., (Smith et al., 2008) need many features and their outcomes are not really understandable by a user. In this case, using them is not satisfactory. Indeed, we are interested in discovering knowledge, which can be easily managed and used in NLP systems in the form of linguistic patterns or rules. One of the strengths is the ability to judge, modify, enhance and improve patterns by a linguistic expert. Although this point is not further addressed here, ultimate understandability makes an important feature of the proposed methodology. Moreover, the method can be straightforwardly applied to any domain without additional effort to develop custom features or hand-crafted rules.

In this paper, we focus on the automated recognition of named entities in general and gene and protein names in particular. Even though this problem has already been tackled by a great range of various methods (see Section 5), it still remains a challenging research issue. We experimentally prove that the pattern mining approach is able to distinguish subtle relationships in text collections to highlight named entities. Sequential patterns [also referred to as sequences in Srikant and Agrawal (1996)] and sequential rules [also referred to as episode rules in Mannila et al. (1997)] are the basis of pattern mining techniques from texts because they take into account the order between the elements of texts. For text entity recognition, the experiments carried out in Section 2 show that sequences can provide suitable scores in recall whereas sequential rules show higher precision. Using only sequential patterns or only sequential rules are not enough to get sufficient recall and precision scores. Our key idea in this paper is to take benefit from synergic action of pattern and rule mining techniques. Patterns can hit a large spectrum of potentially interesting phrases while rules bring necessary precision. This synergy is

further reinforced by simultaneous application of itemset and sequential mining. Although texts must primarily be treated as sequences (of words) otherwise a large portion of information is lost, a pertinent disregard for word order can clarify the context of the core sequence. Figure 1 organises the four affined mining tasks. Despite their common grounds, we are not aware of any other work that combines their strengths in the way we do.

Figure 1 Sequential and non-sequential mining tasks



We propose a generic approach to automatically discover IE rules for the named entity recognition (NER) problem. Our main contribution is to define a method to automatically derive suitable patterns recognising gene and protein names. For that purpose, we have designed a new kind of patterns, left-sequence-right (LSR) patterns taking into account the surrounding context of a sequence and relaxing the order constraint around the sequence. These patterns provide a way to contextualise and model the neighbourhood around a sequence. They exploit the main strength of both sequences and sequential rules. Our approach is entirely automatic so that various texts, including their updates, can be handled. Furthermore, it can be applied to raw text and the discovered rules can easily be understood by the end-users.

2 Motivating example

Biomedical NER aims at identifying the boundary of a substring and then mapping the substring to a predefined category (e.g., gene or disease). Having a training corpus in which named entities are tagged, our goal is to automatically learn extraction rules that can then be applied to untagged text in order to discover named entities.

Table 1 is an example of tagged sentences that we examine in order to discover extraction rules. In these sentences, named entities are tagged in bold with surrounding ⟨…⟩. In this example, we focus on the discovery of gene names. In this section, we show that using pattern mining techniques is promising to automatically discover extraction rules of gene names.

Table 1 An example of tagged sentences from BioCreative corpus

s_1	Comparisons of the four operon control regions studied indicate that the ⟨**NarL heptamers**⟩ are arranged with diverse orientations and spacing.
s_2	Hydroxypropyl methacrylate, a new water-miscible embedding medium for electron microscopy.
s_3	⟨**Tctex-1**⟩ binding required the first 19 amino acids of Fyn and integrity of two lysine residues within this sequence that were previously shown to be important for Fyn interactions with the immunoreceptor tyrosine-based ⟨**activation motifs**⟩ of ⟨**lymphocyte Ag receptors**⟩.
s_4	Closure of an open high below-knee guillotine amputation wound using a skin-stretching device.

Prior to pattern mining application, linguistic preprocessing tasks must be carried out. The corpus has to be tokenised and then it can be stemmed. There are works devoted to this issue such as Schmid (1994). In this paper, we do not focus on the preprocessing of the corpus and we use corpus sentences that are already tokenised. All substrings that are tagged as gene names are labelled with a unique label *AGENE* as shown in Table 2.

Table 2 Transformed sentences to support pattern mining techniques

s_1	Comparisons of the four operon control regions studied indicate that the AGENE are arranged with diverse orientations and spacing.
s_2	Hydroxypropyl methacrylate, a new water-miscible embedding medium for electron microscopy.
s_3	AGENE binding required the first 19 amino acids of Fyn and integrity of two lysine residues within this sequence that were previously shown to be important for Fyn interactions with the immunoreceptor tyrosine-based AGENE of AGENE.
s_4	Closure of an open high below-knee guillotine amputation wound using a skin-stretching device.

It should be noticed that the order of the words within the sentence is primordial in the NER problem since we want to discover boundaries that delimit named entities. The order relation that we considered is the order of the tokens within the sentences. A text sentence is thus seen as a sequence of tokens or stemmas.

The discovery of association rules cannot be straightforwardly applied in this problem because such rules do not take order relation into account. That is why we use sequence-based pattern mining techniques. As preliminary experiments, we applied two pattern mining techniques:

- *Sequential pattern mining* to discover sequences that contain at least one token *AGENE* and that frequently occur in the data with respect to a frequency constraint called *minsup* (minimum support threshold, where the support is simply the number of sentences in which the patterns appear). We can then try to match these specific patterns to the text sentences in order to discover gene names. As an example, the discovered sequence $\langle w_1, w_2, AGENE, w_3, w_4 \rangle$ can then be applied in texts. If $\langle w_1, w_2 \rangle$ and $\langle w_3, w_{4,} \rangle$ are matched in a sentence, then the piece of sentence between w_1, w_2 and w_3, w_4 is tagged as a gene name.

- *Sequential rule mining* considering an additional constraint called confidence threshold that enables us to discover implications (rules) between elements within the sequences. Thus, discovered rules must satisfy both conditions: minimum support threshold and minimum confidence threshold. Confidence of a rule $X \rightarrow Y$ can be interpreted as an estimate of the probability $P(Y \mid X)$, the probability that a sentence, containing X contains also Y after X. The confidence threshold draws the difference between sequential patterns and sequential rules. Indeed, a sequential pattern is a frequent pattern but no interrelation between elements of the sequence is measured. As an example, the sequence *the AGENE* can be frequent on the dataset. However, there is no implication between *the* and *AGENE*. Indeed, many words different from *AGENE* appear after *the* in texts. So, the confidence threshold is not likely to be satisfied for the rule *the* $\rightarrow$ *AGENE*. While if *AGENE* appears nearly all the times after the sequence of words *the overexpression of*, then we could expect to have the rule *the overexpression of* $\rightarrow$ *AGENE* satisfying the confidence threshold. In the NER problem, the purpose is to discover rules where the if-part is a sequence of tokens and the then-part is the special token *AGENE*. These rules enable us to identify the left context of a gene name. By inverting the order relation, other rules can be inferred and the right context can also be identified. Then a pair of rules can be applied to detect the presence of a named entity and then to define its left and right boundaries. For instance, $R_l = \langle w_1, w_2, w_3 \rangle \rightarrow A\,GENE$ and $R_r = A\,GENE \leftarrow \langle w'_1, w'_2, w'_3 \rangle$ can be matched to the sentence $...w_1\,w_2\,w_3\,X\,Y\,Z\,w'_3\,w'_2\,w'_1...$ where *XYZ* is then tagged as a gene name.

In order to define extraction rules that can be applied in text, we put some time constraints on the sequential patterns and sequential rules that we want to mine. Indeed, we want to discover frequent sequences of contiguous words to use the discovered patterns and rules as regular expressions in text.

To measure the relevancy of sequential patterns and sequential rules for the NER problem, we performed experiments on three different datasets. We used two well-known corpora from the literature that have frequently been used as benchmark in several papers and challenges: *GeneTag* from *Genia* dataset by Tanabe et al. (2005) and *BioCreative* dataset from Yeh et al. (2005) (the best F-score for gene/protein name extraction on these corpora are respectively 77.8% and 80%). Furthermore, we consider a very large corpus to fully benefit from scalability of the proposed pattern mining techniques. This corpus, called *Abstracts*, clearly demonstrates that this work handles very large datasets. It contains a set of 35,192 abstracts (305,192 sentences, 44.2 MB of data) collected automatically from NCBI website (http://www.ncbi.nlm.nih.gov). It is a raw text in which each abstract can be seen as a paragraph, the gene and protein name occurrences have been automatically annotated. 228,985 sentences contain at least one gene/protein name. The annotation process is a straightforward projection of terms from a dictionary, which has been learned by Charnois et al. (2006).

We separately applied sequential pattern and rule mining techniques to recognise gene and protein names in these three corpora. For the evaluation, we used a ten-fold cross-validation to partition each initial dataset in a training set and in a testing set. Unfortunately, these techniques did not lead to good results for NER problems as the following experiments show:

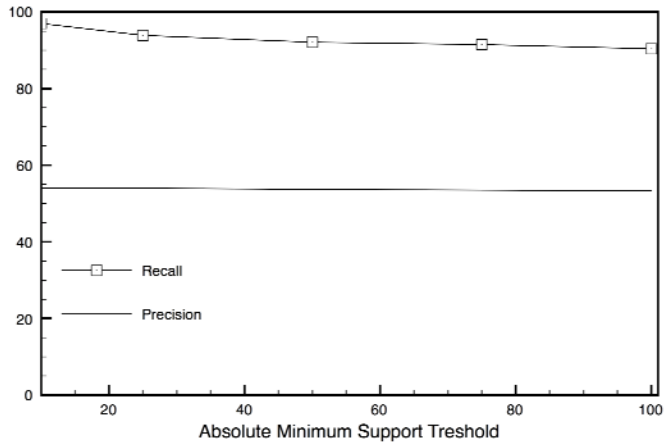
- Graphs from Figures 2(a), 3(a) and 4(a) report the recall and the precision of sequential patterns for gene recognition on the different datasets. We can see that the sequential pattern technique provides very good recall results. Sentences that contain a named entity are widely covered by these patterns. However, these patterns suffer from a lack of precision. Indeed, they cause too many false positives. In numerous cases, sequential patterns match with sentences that do not contain a named entity and then unfortunately identify a word or a group of words as a named entity. As an example, the sequential pattern ⟨*AGENE expression*⟩ enables us the discovery of many gene names but it also engenders the detection of false positives like '*this gene*' or '*the*' in sentences containing '*this gene expression*' or '*the expression*'.
- Graphs from Figures 2(b), 3(b) and 4(b) report the recall and the precision of sequential rules for gene recognition on the different datasets. These curves show that the sequential rule technique provides a good precision by virtue of the confidence measure but the recall is too low. Indeed, discovered sequential rules do not correctly cover sentences that contain a named entity in Figures 2(b) and 3(b). It is due to the fact that many rules are not taken into account because they do not respect the confidence threshold. Note that the precision in Figure 3(b) is not defined when absolute support threshold is set to 50; indeed, the recall is equal to 0% in this case. The third corpus *Abstracts* [Figure 4(b)] shows a different behaviour as it is automatically annotated and the annotation is known to capture the regular gene name occurrences while irregular ones might be omitted. Consequently, the number of false negatives is likely higher than the two other corpora.

It should be noticed that the precision rate seems to stay stable when the support threshold changes except when recall becomes equal to 0% as in Figure 3(b). It is explained by the definition of the precision rate $\left(P_r = \frac{TP}{TP + FP} \right)$: when the support threshold becomes lower, the recall rates increase. So, the number of true positives increases but the number of false positives increases as well. As a consequence, the ratio $\frac{TP}{TP + FP}$ cannot be straightforwardly altered by changes of the support threshold.

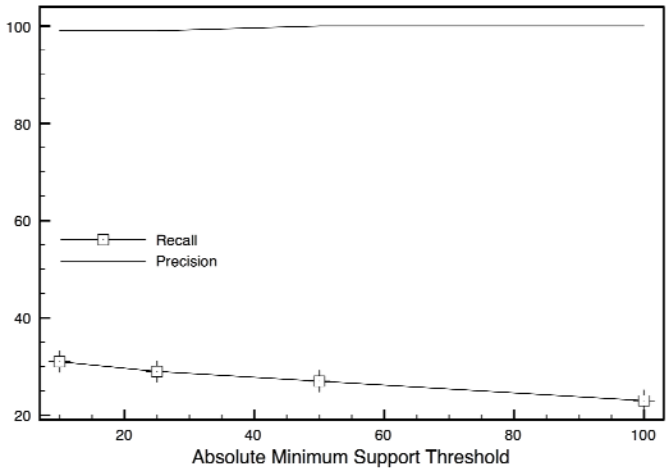
These experiments clearly show the limitations of frequent pattern mining technique for the problem of NER. On the one hand, sequential patterns offer a good coverage of sentences that contain a named entity (high recall) but lead to the detection of too many false positives (low precision). On the other hand, sequential rules provide high precision scores but too low recall. It would be very interesting to make a trade-off between the high precision of sequential rules and the high recall of sequential patterns and to profit from advantages from these kinds of patterns without their limitations. From this empirical study, we propose in this paper the LSR patterns that aim at characterising a sequence by its neighbourhood. LSR patterns combine sequential pattern mining and itemset mining by relaxing the order constraint around frequent sequential patterns. The surrounding context can then be used to check the relevancy of the pattern and thus, reduce the detection of false positives while taking advantage of the good coverage of sequential patterns.

In the rest of the paper, we define LSR patterns and describe how to mine such patterns. We also show how to use them in NER problems.

Figure 2 (a) sequential pattern mining (b) sequential rule mining $\left(minconf = 0.75\right)$ for the NER problem according to *Genia* dataset

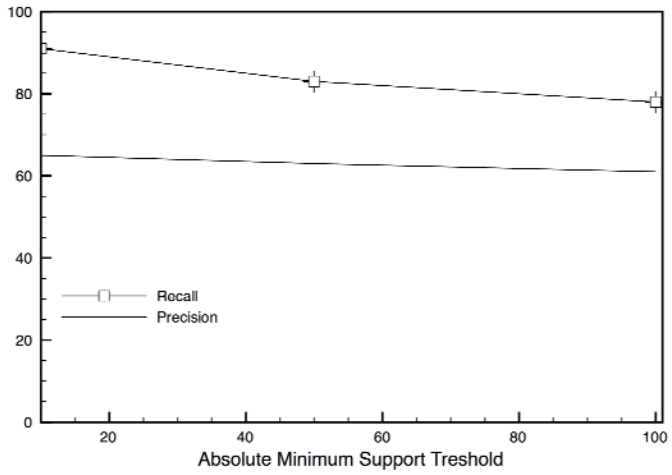


(a)

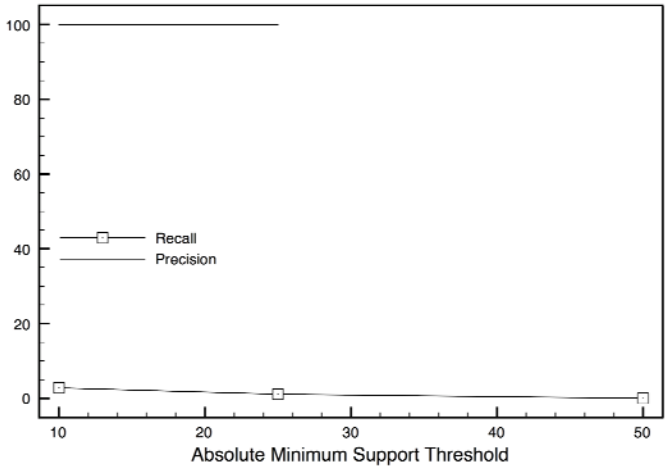


(b)

Figure 3 (a) sequential pattern mining (b) sequential rule mining $\left(minconf = 0.75\right)$ for the NER problem according to *BioCreative* dataset

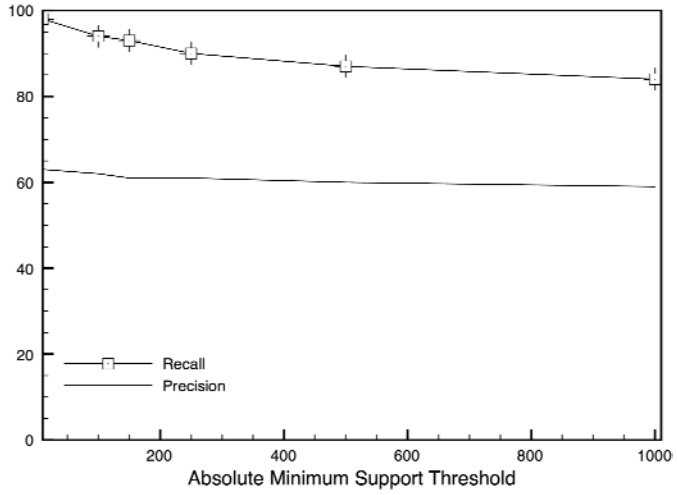


(a)

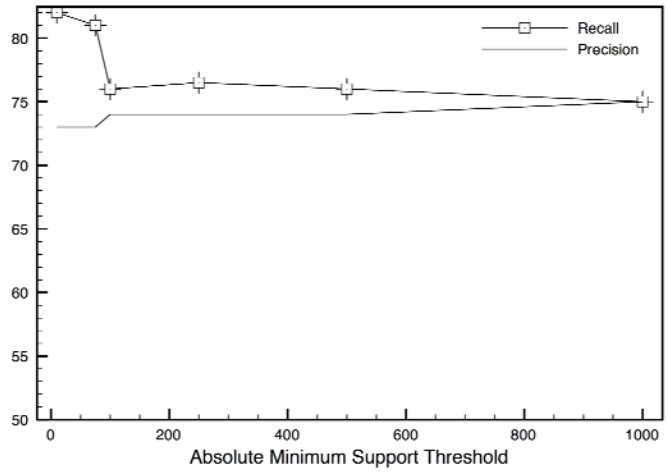


(b)

Figure 4 (a) sequential pattern mining (b) sequential rule mining $(minconf = 0.75)$ for the NER problem according to *Abstracts* dataset



(a)



(b)

3 Our proposal: LSR patterns

Before defining LSR patterns, their extraction and their application to NER problems, let us introduce some preliminary concepts related to sequences and constraints.

3.1 Preliminary definitions

Let $\mathcal{I} = \{e_1, e_2, ..., e_n\}$ be a set of *items*. A *sequence* $s = \langle i_1, i_2, ..., i_k \rangle$ is an ordered list of items. A *sequential pattern* is simply a sequence. A *data sequence* S is a sequence with

time stamps associated to each of its elements. More precisely, a data sequence S is a list $\langle(t_1, j_1),(t_2, j_2),...,(t_m, j_m)\rangle$ where $t_1 < t_2 < ... < t_m$ are time stamps and $j_1, j_2, ..., j_m$ are items. Given a sequential pattern $s = \langle i_1, i_2, ..., i_k \rangle$, a data sequence $o = \langle(u_1, i_1),(u_2, i_2),...,(u_k, i_k)\rangle$ is an occurrence of s in a data sequence S if all elements of o are in S. For instance, $\langle(1,a),(4,b)\rangle$ is an occurrence of the sequential pattern $s = \langle a,b\rangle$ in data sequence $S = \langle(1,a),(2,c),(4,b),(6,b)\rangle$.

A sequence database SDB is a set of tuples (sid, S) where sid is a sequence-id and S a data sequence. A data sequence S is said to *contain* a sequential pattern s, if s has at least one occurrence in S. The *support* of a sequential pattern s in a sequence database SDB is the number of data sequences of SDB that contain s.

Given a minimum support threshold $minsup$, the goal of mining sequential patterns on a sequence database SDB is to find the complete set of sequences whose support is greater than or equal to $minsup$.

Pattern mining involves different challenges, such as designing efficient tools to tackle large datasets and to select patterns of potential interest. The constraint-based pattern mining framework is a powerful paradigm to discover new highly valuable knowledge (see Ng et al., 1998). Constraints allow user to focus on the most promising knowledge by reducing the number of extracted patterns to those of potential interest. There are now generic approaches to discover patterns and sequential patterns under constraints (e.g., De Raedt et al., 2002; Soulet and Crémilleux, 2005; Pei et al., 2002; Garofalakis et al., 1999; Leleu et al., 2003). Note that constraint-based pattern mining challenges two major problems in pattern mining: effectiveness and efficiency. Indeed, mining may lead to knowledge flooding with patterns uninteresting to users and it often takes substantial processing power for mining the complete set of patterns in large databases. So, constraints can be used to enhance both the quality of discovered patterns and the mining process.

Let constraint C for a sequential pattern s be a Boolean function $C(s)$. A set of constraints $\mathcal{C} = \{C_1, C_2, ..., C_n\}$ for a sequential pattern s is then the conjunction of all Boolean functions $C_i(s)$ from $\mathcal{C}$. Then, given a set of constraints $\mathcal{C}$, the problem of constraint-based sequential pattern mining is to find the complete set of sequential patterns satisfying every condition C_i from $\mathcal{C}$.

Note that even if the support condition is a constraint, it does not belong to $\mathcal{C}$. Indeed, pattern mining is based on this key condition and $\mathcal{C}$ models *additional constraints different from frequency constraint*. There are various types of constraints such as *syntactic*, *length*, *duration* and *gap constraints*, see Pei et al. (2002).

3.2 LSR pattern: a new kind of pattern

As we have noticed, sequences and sequential rules present some non-negligible limitations for NER problem in biomedical data. In order to take advantage of the high recall of sequences and improve their precision, we propose a new type of pattern called LSR pattern. They enable us to characterise a sequence with itemsets representing its

surrounding context. Indeed, the key idea is to relax the order constraint around a sequence in order to model left and right neighbourhood of a sequence, thanks to itemsets.

Definition 3.1: (LSR) A LSR pattern x is a triplet $x=(l,s,r)$ where:

- s is a sequential pattern
- l and r are sets of items.

LSR patterns go further than the result of a combination between a sequence and two itemsets. Indeed, such patterns provide a way to contextualise a sequence, thanks to its neighbourhood. Thus, itemsets l and r provide a way to model neighbourhood around sequence s. As an example, consider an LSR pattern $x_1=(\{\},\langle the, AGENE,\rangle\{gene, with, associated\})$ where $l=\{\}$ and $r=\{gene, with, associated\}$ which means that these words are in the right neighbourhood of the sequence *the AGENE*.

The order relation constraint is relaxed around frequent sequential patterns in data sequences in order to extract frequent itemsets that model the neighbourhood of the sequence and contextualise it in the data sequences. To formalise the extraction of frequent LSR patterns, we need to introduce the following definitions.

Contrary to an itemset that occurs at most once in a transaction in the itemset mining problem, a sequence may appear several times in a data sequence (see example below). Consequently, for a same data sequence, there are different ways to identify the neighbourhood of a sequence within the data sequence. In order to exhibit the *most representative* itemsets that model neighbourhood, we introduce the notion of 'compact *occurrence of* s *in a data sequence*'.

Definition 3.2: (compact occurrence) Given a sequential pattern $s=\langle i_1,i_2,...,i_k\rangle$, a set of constraints $\mathcal{C}$ and a data sequence S, then an occurrence o_c of s in S, where $o_c=\langle(t_1,i_1),(t_2,i_2),...,(t_k,i_k)\rangle$ is a compact occurrence of s in S if the following conditions hold:

- o_c satisfies $\mathcal{C}$
- there is no occurrence $o'=\langle(t'_1,i_1),(t'_2,i_2),...,(t'_k,i_k)\rangle$ of s in S such that $o'\neq o_c$ and o' satisfies $\mathcal{C}$ and $t_1\leq t'_1$ and $\forall\alpha\in\{2,...,k\}, t'_\alpha\leq t_\alpha$.

This definition enables to focus on the minimal pieces of the data sequence S that contain the sequence s. Indeed, a data sequence S can contain several compact occurrences of a sequence s. Compact occurrences have similar semantics to 'minimal occurrences' from Mannila et al. (1997).

As an example, given $\mathcal{C}=\emptyset$, the data sequence $S=\langle(1,a),(2,c),(3,b),(4,d),(5,a),(7,a),(8,b),(10,e),(12,f),(14,a),(15,g),(16,h),(18,b),(20,c)\rangle$ contains three different compact occurrences of $s=\langle a,b\rangle$:

1 $\langle(1,a),(3,b)\rangle$

2 $\langle(7,a),(8,b)\rangle$

3 $\langle(14,a),(18,b)\rangle$.

Note that $\langle(1,a),(18,b)\rangle$ is not a compact occurrence of s in S since it is not minimal.

If we add a maximal gap constraint $max\_gap = 2$ to $\mathcal{C}$ meaning that the maximal time gap between consecutive items of the sequence s is 2, then S contains two compact occurrences of s : $\langle(1,a),(3,b)\rangle$ and $\langle(7,a),(8,b)\rangle$.

Since a data sequence can contain several compact occurrences of s, we speak of the ith *compact occurrence* of s in S (denoted by o_c^i) where i refers to order of appearance of the compact occurrence within the data sequence. According to the previous example, where $\mathcal{C}=\emptyset, \langle(1,a),(3,b)\rangle$, $\langle(7,a),(8,b)\rangle$ and $\langle(14,a),(18,b)\rangle$ are respectively the first, second and third compact occurrences of s in S.

In order to define a way to identify neighbourhood with itemsets, we have to define the notion of the prefix of a ith compact occurrence.

Definition 3.3: (prefix of an occurrence) Let o_c^i be the ith compact occurrence of s in S, the prefix of o_c^i in S is equal to the subsequence of S starting at the beginning of S and ending strictly before the first item of o_c^i.

In our example, where $S = \langle(1,a),(2,c),(3,b),(4,d),(5,a),(7,a),(8,b),(10,e),(12,f),(14,a),(15,g),(16,h),(18,b),(20,c)\rangle$ is the input sequence, $s = \langle a,b\rangle$ and $\mathcal{C}=\emptyset$, we have:

- the prefix of the first compact occurrence o_c^1 of s in S is equal to $\langle\rangle$
- the prefix of o_c^2 is equal to $\langle(1,a),(2,c),(3,b),(4,d),(5,a)\rangle$
- the prefix of o_c^3 is equal to $\langle(1,a),(2,c),(3,b),(4,d),(5,a),(7,a),(8,b),(10,e),(12,f)\rangle$

In the same way, we introduce the notion of suffix of a ith compact occurrence in a data sequence.

Definition 3.4: (suffix of an occurrence) Let o_c^i be the ith compact occurrence of s in S, the suffix of o_c^i in S is the subsequence of S starting just after the last item of o_c^i to the end of S.

According to our example where $s = \langle a,b\rangle$ and $\mathcal{C}=\emptyset$, we have the following suffixes:

- the suffix of the first compact occurrence o_c^1 of s in S is equal to $\langle(4,d)(5,a),(7,a),(8,b),(10,e),(12,f),(14,a),(15,g),(16,h),(18,b),(20,c)\rangle$
- suffix of o_c^2 is equal to $\langle(10,e),(12,f),(14,a),(15,g),(16,h),(18,b),(20,c)\rangle$
- suffix of o_c^3 is equal to $\langle(20,c)\rangle$.

In order to delimit the range of the neighbourhood around compact occurrences, we introduce a parameter N_R to consider only items having time-stamps sufficiently close to compact occurrences (absolute difference between item time-stamp and time-stamp of the closest element of a compact occurrence must not be greater than N_R). This constraint is taken into account in the prefix and the suffix of the ith compact occurrence of s in S. Indeed, only items which respect neighbourhood range N_R to o_c^i are returned.

According to our example, given $s=\langle a,b\rangle$, $\mathcal{C}=\emptyset$ and $N_R=5$:

- $prefix\left(o_c^i,S,N_R\right)=\langle\ \rangle$ and $suffix\left(o_c^i,S,N_R\right)=\langle(4,d),(5,a),(7,a),(8,b)\rangle$
- $prefix\left(o_c^2,S,N_R\right)=\langle(2,c),(3,b),(4,d),(5,a)\rangle$ and $suffix\left(o_c^2,S,N_R\right)=\langle(10,e),(12,f)\rangle$
- $prefix\left(o_c^3,S,N_R\right)=\langle(10,e),(12,f)\rangle$ and $suffix\left(o_c^3,S,N_R\right)=\langle(20,c)\rangle$.

Note that N_R can be automatically set by studying the average size of the prefix and the suffix of compact occurrences.

Definition 3.5: (inclusion of LSR pattern) Given N_R and a set of constraints $\mathcal{C}$, a LSR pattern $x=(l,s,r)$ is included in a sequence S if the following conditions held:

1 s has a compact occurrence in S

2 $\exists i$ such that $\forall_{el}\in l$, item e_l appears in $prefix\left(o_c^i,S,N_R\right)$ and $\forall_{er}\in r$, item e_r appears in $suffix\left(o_c^i,S,N_R\right)$, where o_c^i is the ith compact occurrence of s in S.

To support a LSR (l,s,r) pattern, a data sequence first must contain the sequential pattern s. Then, it must exist o_c^i, an ith compact occurrence of s in S such that all elements of l must be contained in the prefix of o_c^i with respect to N_R. Moreover, for the same compact occurrence o_c^i, all elements of r must also be contained in the suffix of o_c^i with respect to N_R. Note that the order constraint is relaxed for l and r. Indeed, elements from these itemsets must be contained in the neighbourhood of the sequence, whatever their order of appearance.

According to the previous definition, we can define the support of a LSR pattern in a sequence database.

Definition 3.6: (Support) Given a set of data sequences SDB and a neighbourhood range N_R, the support of a LSR pattern x is the number of sequences from SDB that contain x.

The problem of mining LSR patterns aims at discovering *frequent* LSR patterns from a sequence database. In order to avoid some redundancies, we return frequent LSR patterns having maximal itemsets.

Definition 3.7: (LSR pattern mining problem) Let SDB be a set of data sequences and N_R be a radius of neighbourhood. Given a minimum support threshold $minsup$, the problem of mining LSR patterns is to find the complete set of LSR patterns FS from SDB defined as the set $FS = \{x = (l,s,r)\ s.t.\ support(x) \geq minsup$ and $\nexists x' = (l',s,r')$ having $support(x') \geq minsup$ where $l \sqsubseteq l'$ and $r \sqsubseteq r'$ and $x \neq x'\}$.

The problem of mining LSR patterns is difficult since it combines constraint based sequence mining and itemset mining when the order constrain is relaxed around a frequent sequence within data sequences. Nevertheless, the next section shows how we overcome this difficulty and it provides our method to mine LSR patterns.

3.3 LSR pattern mining algorithm

Our method to extract LSR pattern is divided into two constraint-based mining steps. First, the set $SAT(\mathcal{C})$ of sequential patterns that satisfy the set of constraints $\mathcal{C}$ is discovered from SDB. Then, a new set SDB' of data sequences is generated according to patterns from $SAT(\mathcal{C})$. The LSR patterns are then extracted from this dataset SDB'.

Algorithm 1 describes the extraction of frequent LSR patterns. Let us describe more precisely its different steps.

Given SDB, $minsup$, and $\mathcal{C}$, the first step of the algorithm is to find the set of sequential patterns in SDB that satisfy $\mathcal{C}$, denoted $SAT(\mathcal{C})$.

Then, the algorithm transforms SDB into a new sequence dataset SDB' according to $SAT(\mathcal{C})$. It builds a set of identifiers $\mathcal{P}_{id}$ associated to the patterns in $SAT(\mathcal{C})$, that will be used as additional items in the new dataset. For each occurrence o_c of the patterns in $SAT(\mathcal{C})$ (first loop), a new sequence S' is built. In such a sequence S', a pattern identifier replaces the occurrence o_c, and on the left and on the right of the occurrence only the elements within the N_R neighbourhood are conserved. As an example, given a neighbourhood $N_R = 4$, and a sequential pattern $s = \langle a,b,c \rangle$, from its compact occurrence $\langle (3,a),(6,b),(9,c) \rangle$ in the data sequence $S = \langle (1,a),(2,c),(3,a),(4,d),(6,b)\ (8,d),(9,c),(11,a),(12,d),(14,e),(18,c) \rangle$ of SDB, the algorithm generates in SDB' the sequence $S' = \langle (1,a),(2,c),(3,pattId(s)),(11,a),(12,d) \rangle$.

Algorithm 1 LSR pattern mining

Data: Sequence database SDB, minimum support threshold $minsup$, set of constraints $\mathcal{C}$, neighbourhood range N_R

Result: Set of frequent LSR patterns

begin

$SAT(\mathcal{C}) \leftarrow$ FrequentSequenceMining$(minsup, SDB, \mathcal{C})$;

$SDB' \leftarrow \emptyset$;

Associate to each pattern s in $SAT(\mathcal{C})$ a new symbol denoted $pattId(s)$;

Let $\mathcal{P}_{id}$ be the set of all these new symbols;

for each compact occurrence o_c of the patterns in $SAT(\mathcal{C})$ do

Let o_c be of the form: $\langle (t_1,i_1),(t_2,i_2),...,(t_k,i_k)\rangle$;

Let S be the data sequence where o_c, occurrence of a pattern s, has been found;

$S' \leftarrow prefix(o_c, S, N_R) \oplus \langle (t_1, pattId(s))\rangle \oplus suffix(o_c, S, N_R)$

// where $\oplus$ denotes list concatenation

$SDB' \leftarrow SDB' \cup \{S'\}$

$\mathcal{C}' \leftarrow \{$pattern must contain an element of $\mathcal{P}_{id}\}$

$SAT(\mathcal{C}') \leftarrow$ FrequentSequenceMining$(minsup, SDB', \mathcal{C}')$;

$\mathcal{R} \leftarrow \emptyset$;

for each pattern p in $SAT(\mathcal{C}')$ do

Let p be of the form: $\langle i_1, i_2, ..., i_n, id, i'_1, i'_2, ..., i'_m\rangle$ where $id \in \mathcal{P}_{id}$;

Let s be the sequential pattern such that $pattId(s) = id$;

Left $left$ be the set of the different items appearing in $i_1, i_2, ..., i_n$;

Left $right$ be the set of the different items appearing in $i'_1, i'_2, ..., i'_m$;

$\mathcal{R} \leftarrow \mathcal{R} \cup \{\langle left, s, right\rangle\}$;

Remove from $\mathcal{R}$ the LSR patterns that are not maximal;

return $\mathcal{R}$;

end

Next, from SDB', the algorithm extracts, the sequential patterns that contain an identifier of one of the patterns extracted from SDB. Then (second loop), for each pattern p obtained from SDB', the algorithm retrieves the identifier part $(id \in \mathcal{P}_{id})$ to find the corresponding sequential pattern s extracted from SDB. This pattern s forms the central part of a LSR pattern. The algorithm takes the different items on the left (resp. on the right) of the id to form the left (resp. right) part of this LSR pattern. Finally,

non-maximal patterns are removed from the resulting set $\mathcal{R}$, where $x = \langle (l, s, r) \rangle$ is non-maximal if there is another LSR pattern $x' = \langle (l', s, r') \rangle$ such that $x' \neq x$ and $l \sqsubseteq l'$ and $r \sqsubseteq r'$.

The algorithm is based on two sequence mining steps. It is complete because the sequence mining algorithm, which is used, is complete.

3.4 Use of LSR patterns for NER problems

LSR patterns can be used for biomedical NER problem. To challenge this problem, we have first to extract specific LSR frequent patterns. Then, we have to correctly use this set of LSR patterns for discovering named entities in natural language texts.

3.4.1 Extracting LSR patterns for NER problems

First, we have to mine frequent LSR patterns on a tagged and tokenised corpus with special constraints. Sequences must contain a biomedical named entity. As an example, sequences must contain an item *AGENE* in the case of gene name recognition. Moreover, a time constraint is added in order to consider only consecutive events. This constraint is primordial for the use of sequences as regular expression in the recognition phase.

To extract patterns in the experiments presented in this paper, we used our own prototype implemented in C and called *dmt4sp*. This program enables to extract patterns that encompass substring patterns, serial episodes from Mannila et al. (1997) and a limited form of sequential patterns (see Agrawal and Srikant, 1995). It performs complete extractions of the patterns in a collection of sequences, under a combination of constraints on the support and syntax of the patterns, and on the time intervals between the events. The support constraint includes both the support in number of occurrences of the patterns (as defined by Mannila et al., 1997), and the support in number of sequences containing at least one occurrence of the patterns (as defined by Agrawal and Srikant, 1995). The second kind of constraints, the syntactic constraints, includes constraints on the prefix of the patterns and on the pattern sizes (minimum and maximum size). Finally, the time interval constraints enable to set the minimum and maximum time span between events and also between the first and the last element of the patterns. The pattern enumeration method is a standard depth-first prefix-based strategy. It combines constraint checking with a management of occurrences using the occurrence list approach (see Zaki, 2000) with a virtual database projection proposed by Pei et al. (2001), and an efficient handling of multiple occurrences as Meger and Rigotti (2004) and Nanni and Rigotti (2007), under the so-called minimal occurrence semantics from Mannila et al. (1997).

We propose to associate a confidence measure to the sequential pattern s of each LSR pattern. The aim of this measure is to determine if the sequential pattern can be applied on its own or if it is necessary to study its surrounding context (itemsets l and r) to apply it. The confidence of a sequential pattern s for a entity name E is equal to the support of s divided by the support of the sequential pattern s in which the items corresponding to the entity name E (e.g., *AGENE*) have been replaced by a wild-card $*$. This sequential pattern is denoted $s[E/*]$, and definitions of support and occurrence also apply to it, with the wild-card $*$ matching any word or group of words.

Definition 3.8: (Confidence) Given a named entity E, the confidence of a sequential pattern s, containing E, is equal to:

$$Confidence_E(s) = \frac{support(s)}{support(s[E/*])}$$

This measure aims at determining if the occurrence of entity E could be related to the presence of the other items of the sequence. For instance, if the support of a sequence ⟨*the gene AGENE interacts with*⟩ is similar to the support of the sequence ⟨*the gene* ∗ *interacts with*⟩ (confidence $\simeq 1$), this means that when a sentence contains '*the gene*' and further '*interacts with*', there is a gene name between them.

Algorithm 2 Use of LSR pattern for NER problems

Data: Sentence S, LSR pattern $x = (l, s, r)$, minimum confidence threshold $minconf$, neighbourhood range N_R, minimum number of words W_{min}, named entity E

begin

for each compact occurrence o_c of $s[E/*]$ in S do

if $Confidence(s) \geq minconf$ then

Label with E the part of o_c corresponding to $*$ in $s[E/*]$;

else

if $|prefix(o_c, S, N_R) \cap l| + |suffix(o_c, S, N_R) \cap r| \geq W_{min}$ then

Label with E the part of o_c corresponding to $*$ in $s[E/*]$;

else

Do not apply s;

end

3.4.2 Detection of named entities

Algorithm 2 describes how a LSR pattern can be applied or not to a sentence. Given a sentence in natural language, this sentence is tokenised and then we try to find patterns from the set of frequent LSR patterns that can be applied to the tokenised sentence. If a sequential pattern s of a LSR pattern $x = (l, s, r)$ can be applied (all tokens of s that are different from a named entity are perfectly matched), we check the confidence of s. If the confidence is greater than a minimum confidence threshold $minconf$, then, we consider that s can be applied on its own. Otherwise, s is not confident enough to be directly applied. It is thus, necessary to examine its surrounding context. If a sufficient number of items, according to a threshold W_{min}, from l and r match the left and right contexts of s within the tokenised sentence, then the use of s is considered to be relevant, according

to the context. Notice that the sequential pattern s can be applied several times within the sentence. So it is necessary to consider all compact occurrences. Since the number of compact occurrences within a sentence is finite, Algorithm 2 terminates.

4 Experiments

We report experiments performed on real datasets described in Section 2: *BioCreative* (Yeh et al., 2005, cf. Figure 5), *Genia* (Tanabe et al., 2005, cf. Figure 6) and *Abstracts* (cf. Figure 7). These experiments aim at showing the interest of LSR patterns, especially in the biomedical the NER problem where they represent an excellent trade-off between the high-precision of sequential rules and the high recall of sequential patterns. Each corpus was tokenised. According to the previous definitions, each sentence is a data sequence. SDB is then the set of sentences from a corpus. We used a ten-fold cross validation to partition each initial data set in a training set and in a testing set.

LSR patterns excel in exploitation of formerly unconfident sequential patterns. As an example, *that AGENE* is not confident at all, but some words frequently appear in the left neighbourhood of this pattern (*indicated*, *revealed*, *demonstrate*, *evidence*) and in the right neighbourhood (*binds*, *expressed*, *activity*, *protein*, etc.). As a consequence, such unconfident sequential patterns that seem to be useless for the NER problem can be applied, thanks to their neighbourhood.

The goal of the experiments is to evaluate the quality of recognition of LSR patterns for NER problems. We also study the behaviour of LSR patterns according to the minimum support, the minimum confidence and W_{min}. In all experiments, we fix $N_R = 5$ for linguistic reasons, it is a size for which linguists consider that it makes sense to try to connect words.

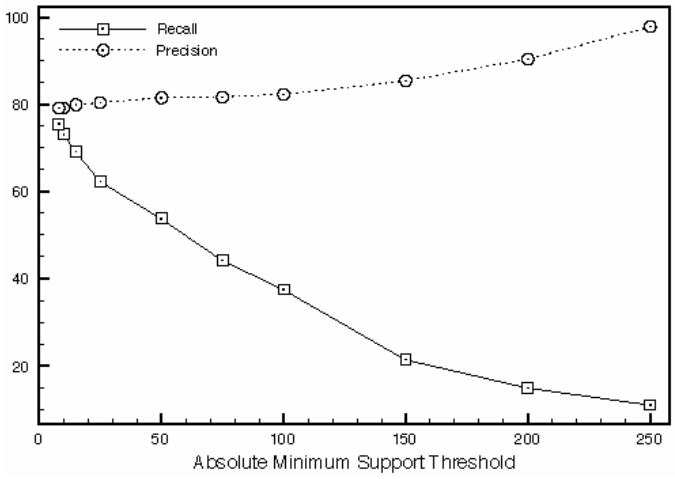
Figures 5(a), 6(a) and 7(a) describe the precision and recall of LSR patterns for NER problems according to the absolute minimum support threshold. The behaviour of LSR patterns is similar in the three plots. The recall increases and the precision decreases when the minimum support threshold becomes smaller. Indeed, there is a larger set of frequent LSR patterns that thus provides a better coverage (better recall) for the detection of named entities. However, this larger set leads also to the detection of a greater number of false positives and then to a lower precision.

Figures 5(b), 6(b) and 7(b) aim at comparing the performance of LSR patterns, sequential patterns and sequential rules for the NER problem. To compare these approaches, we use the well-known F-measure $F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$ (see Van Rijsbergen, 1979) that is the harmonic mean of precision and recall. Indeed, this measure aims to make a trade-off between precision and recall. So we use it to evaluate and compare the performance of the different approaches. Note that there is no result for sequential rules in Figure 5(b) because this technique gave too bad results in this *BioCreative* corpus [see Figure 3(b)]. For *BioCreative* corpus [Figure 5(b)], LSR patterns are significantly better than sequential patterns. On *Genia* corpus [Figure 6(b)], LSR patterns are also better when the minimum support threshold is low. On *Abstracts* corpus [Figure 7(b)], LSR patterns overall give the best F-scores. Note again that sequential rules are better than sequential patterns on this corpus. These different plots show the

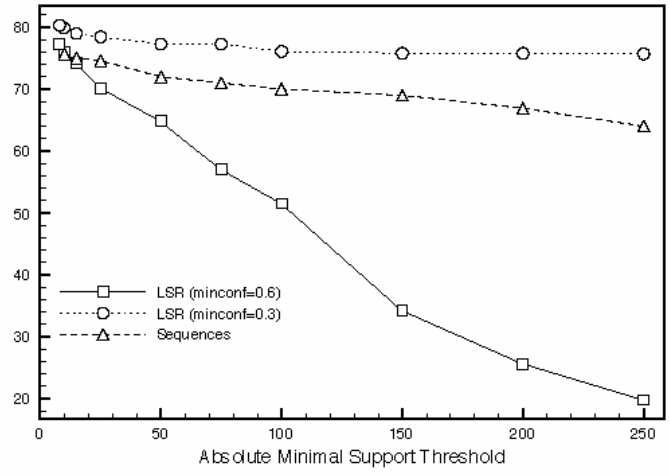
interest of LSR patterns for the NER problem since they overcome sequential patterns and sequential rules.

Figures 5(c), 6(c) and 7(c) report the recall and precision of LSR patterns when the minimum confidence threshold changes. The three plots are similar. The recall increases and the precision decreases when the minimum confidence threshold becomes lower. Indeed, the lower the confidence threshold, the bigger the number of false positives is. However, we can notice that the neighbourhood awareness lead to preserve the good precision of LSR patterns.

Figure 5 Experiment on *BioCreative* dataset, (a) precision and recall of LSR patterns $(W_{min}=3, N_R=5)$ (b) F-score of LSR patterns, sequential patterns $(W_{min}=3, N_R=5)$ (c) precision and recall of LSR patterns according to $minconf\,(minsupp=10, vmin=3, N_R=5)$ (d) precision and recall of LSR patterns according to $W_{min}\,(minconf=0.6, N_R=5)$

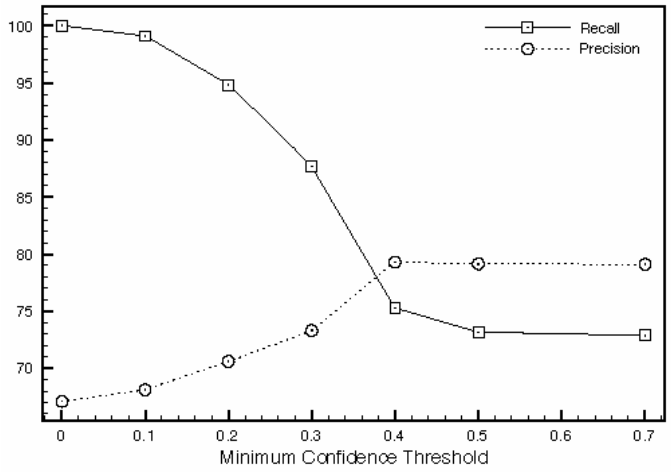


(a)

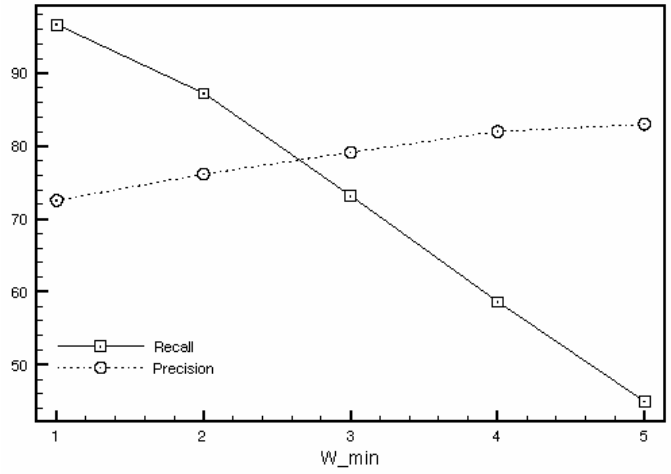


(b)

Figure 5 Experiment on *BioCreative* dataset, (a) precision and recall of LSR patterns $(W_{min}=3, N_R=5)$ (b) F-score of LSR patterns, sequential patterns $(W_{min}=3, N_R=5)$ (c) precision and recall of LSR patterns according to $minconf(minsupp=10, vmin=3, N_R=5)$ (d) precision and recall of LSR patterns according to $W_{min}(minconf=0.6, N_R=5)$ (continued)



(c)

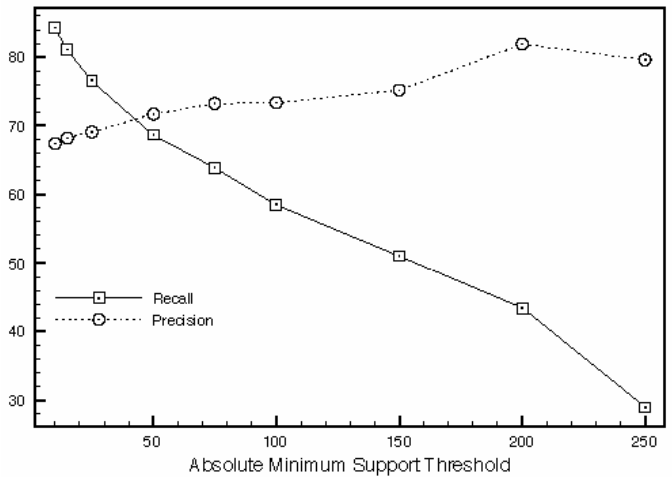


(d)

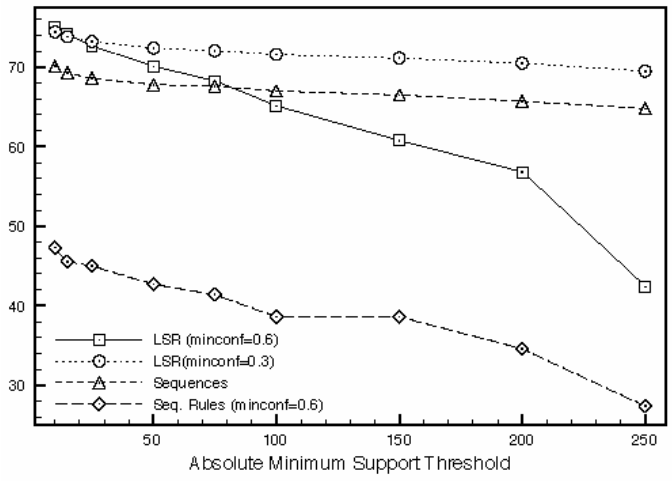
Figures 5(d), 6(d) and 7(d) report the recall and the precision of LSR patterns according to W_{min}. This parameter means that at least W_{min} items from the itemsets l and r must be present in the neighbourhood of the sequential pattern s to take the LSR pattern $x=(l,s,r)$ into account for the detection of a named entity. When W_{min} is too important, it is difficult for LSR patterns to satisfy this condition whereas they easily

satisfy it when W_{min} is small. Therefore, the precision increases and the recall decreases when W_{min} becomes higher.

Figure 6 Experiment on *Genia* dataset, (a) precision and recall of LSR patterns $(W_{min} = 3, N_R = 5)$ (b) F-score of LSR patterns, sequential patterns and sequential rules $(vmin = 3, N_R = 5)$ (c) precision and recall of LSR patterns according to $minconf(W_{min} = 3, N_R = 5)$ (d) precision and recall of LSR patterns according to $W_{min}(minsupp = 50, minconf = 0.6, N_R = 5)$

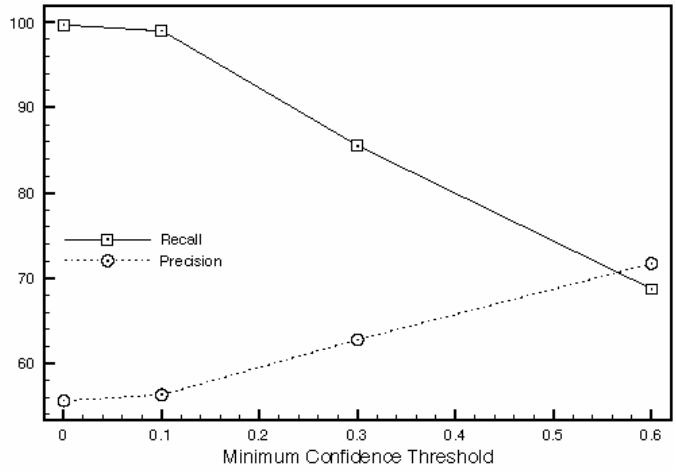


(a)

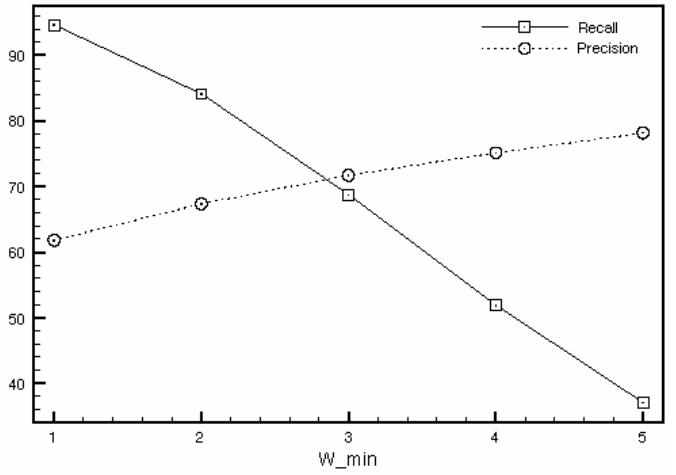


(b)

Figure 6 Experiment on *Genia* dataset, (a) precision and recall of LSR patterns $(W_{min} = 3, N_R = 5)$ (b) F-score of LSR patterns, sequential patterns and sequential rules $(vmin = 3, N_R = 5)$ (c) precision and recall of LSR patterns according to $minconf(W_{min} = 3, N_R = 5)$ (d) precision and recall of LSR patterns according to $W_{min}(minsupp = 50, minconf = 0.6, N_R = 5)$ (continued)

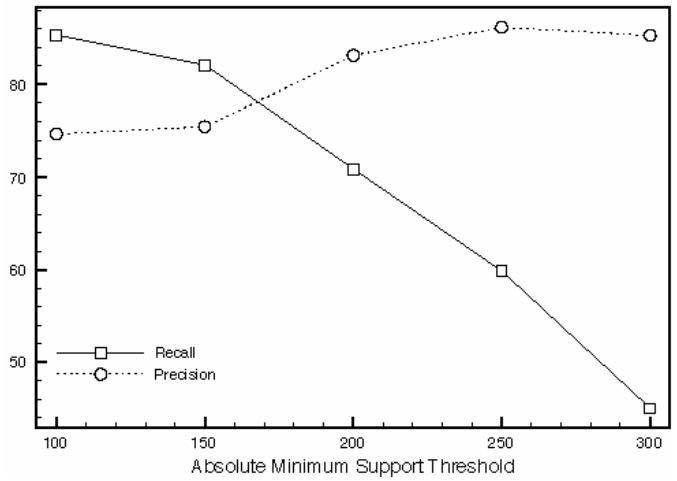


(c)

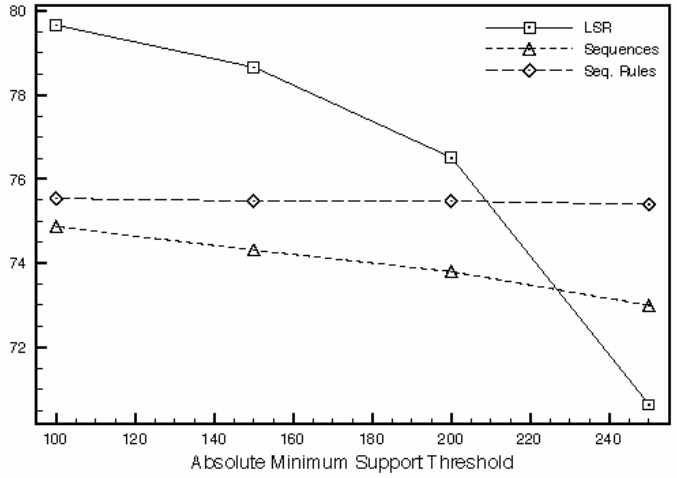


(d)

Figure 7 Experiment on *Abstracts* dataset, (a) precision and recall of LSR patterns $(W_{min}=3, N_R=5)$ (b) F-score of LSR patterns, sequential patterns and sequential rules $(W_{min}=3, N_R=5, minconf=0.6)$ (c) precision and recall of LSR patterns according to $minconf(minsupp=100, W_{min}=3, N_R=5)$ (d) precision and recall of LSR patterns according to $W_{min}(minsupp=100, minconf=0.6, N_R=5)$

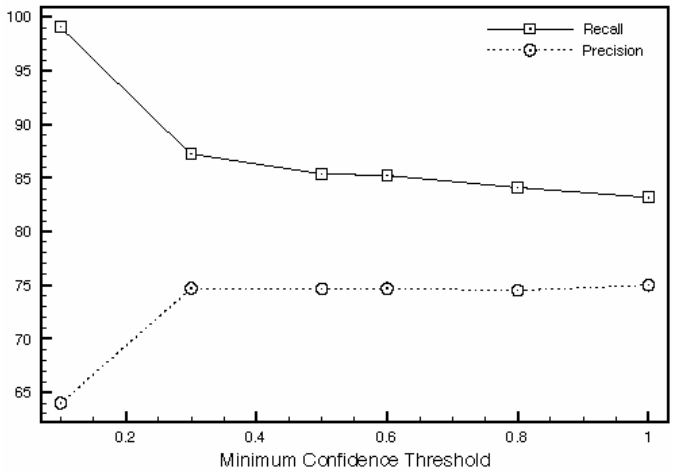


(a)

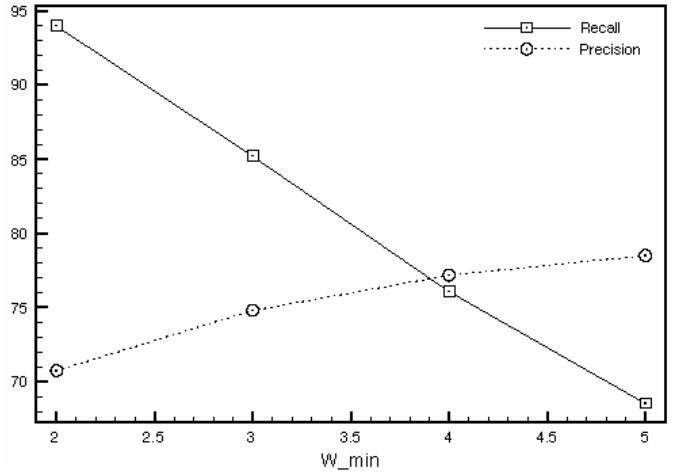


(b)

Figure 7 Experiment on *Abstracts* dataset, (a) precision and recall of LSR patterns $(W_{min}=3, N_R=5)$ (b) F-score of LSR patterns, sequential patterns and sequential rules $(W_{min}=3, N_R=5, minconf=0.6)$ (c) precision and recall of LSR patterns according to $minconf(minsupp=100, W_{min}=3, N_R=5)$ (d) precision and recall of LSR patterns according to $W_{min}(minsupp=100, minconf=0.6, N_R=5)$ (continued)



(c)



(d)

These experiments show the strengths of our approach. Taking the neighbourhood of a sequential pattern into account provides promising results. Indeed, LSR patterns overcome sequential patterns and sequential rules. According to the best results for gene/protein name recognition on *Genia* and *BioCreative* corpora (F-score are respectively 77.8% and 80%), our results are comparable. Moreover, LSR patterns are easily understandable. As an example, we discover the following pattern ⟨{} ⟨*AGENE*, *expression*, *in*⟩,{*cells*}⟩ that means that the word *cells* appears in many cases in the right

neighbourhood of the frequent sequence ⟨*AGENE expression in*⟩. This illustrates another important interest of our approach: the possible use of LSR patterns in a NLP system by a linguistic expert and/or as linguistic resource.

5 Related works

NER is an IE subtask, which consists in locating some strings in a corpus and then assigning a predefined category (gene, protein, biological function) to them. NER has to deal with several difficulties such as polysemy (multi-sense words), synonymy, multi-word terms, variability in the form of names and neologism. It can be considered as a NLP problem and linguistic analysis based methods are one of the proposed approaches in literature such as Cohen and Hunter (2004) and Cohen and Hersh (2005). They are named 'rule-based approaches' since they aim at defining regular expressions, linguistic patterns or grammars that match gene or protein names [for example, Fukuda et al. (1998), one of the earliest systems]. Some of them (e.g., Humphreys et al., 2000) use terminological resources: databases, ontologies, such as UMLS and LocusLink. According to Leser and Hakenberg (2005) rule-based approaches can reach high precision but recall is often low if the rules are too specific making the system not robust enough towards new named entities. Another critical point has to be considered: the rules are manually designed by human experts, are a highly time consuming task and portability is costly.

A second broad category of approaches appeared with the availability of annotated corpus: methods based on ML techniques have been investigated and some promising results have already been obtained by cross-fertilisation of IE and ML techniques on biomedical texts (see Chang et al., 2006; Nédellec et al., 2006 for a review; Smith et al., 2008 for recent systems used during the latest BioCreative challenge, BioCreative II). A large variety of approaches can be used: decision trees, Bayesian classifiers, maximum entropy, hidden Markov models, support vector machines and conditional random fields. Some of the ML approaches use sequence based systems, considering the complete ordered sequences of words in sentences: for example Kinoshita et al. (2005) and Dingare et al. (2004). The first one retrains a dedicated train tagger (TnT-Tagger) by including sequential information, and the second one uses entropy model for predicting the most probable sequence of classifications for words of a sentence. These works often use statistical discriminators and differ from our approach by building models that can only be used as black boxes to perform predictions, but that cannot be interpreted by linguistic or biological experts. For example, SVMs draw a hyperplane in an n-dimensional space, from which deducing readable and understandable patterns is not feasible. However, there are some systems that aim at learning some linguistic rules that can be read and understood by human experts. For instance, Califf and Mooney (1999), Kim et al. (2007) and Cakmak and Özsoyoglu (2007) learned rules in the form of single slot IE patterns or textual extraction patterns, which are equivalent to our sequential patterns (the s in our LSR patterns). These systems have not been used for name gene recognition but for extracting relations between entities [relations for protein/gene annotations in Kim et al. (2007) and Cakmak and Özsoyoglu (2007)], and do not consider the intrinsic issue of high precision/low recall problem that is due to the use of sequential patterns (see Leser and Hakenberg, 2005). In our approach, this problem is overcome by the use of contextual information (the l and r parts of the LSR patterns).

Mining sequential data is not limited to the application presented in this paper and arises in many domains, to analyse various kinds of data including customer transactions, web logs, geophysical data, medical data and of course biological sequences. Most of the time, due to the size of the datasets and to the size of the pattern space, mining sequential data is a difficult task. It has received a lot of attention in the literature, from the extraction of substrings (e.g., Ukkonen, 1995) to the extraction of more general patterns like sequential patterns (e.g., Agrawal and Srikant, 1995) and episodes (e.g., Mannila et al., 1997). One of the most salient extensions of these techniques is the use of constraints, to focus on the patterns of interest, together with the active use of these constraints to reduce the search space (e.g., Srikant and Agrawal, 1996; Zaki, 2000; Garofalakis et al., 1999; Lee and De Raedt, 2004), and to improve the efficiency of the extractions in practice. Pinto et al. (2001) and Stefanowski and Ziembinski (2005) try to contextualise sequential patterns. However, LSR patterns are different from these context-based sequential patterns. Indeed, Pinto et al. (2001) and Stefanowski and Ziembinski (2005) aim at using a set of attributes to characterise a sequential pattern. The attributes that contextualise sequential patterns do not appear within the sequential patterns whereas neighbourhoods and sequences of LSR patterns are described with the same set of attributes.

6 Conclusions

In this paper, we introduced a new type of pattern for sequential data mining, the LSR pattern. It benefits from synergic action of sequential pattern and rule mining as well as frequent itemset mining. It aims to characterise a sequential pattern by its surrounding context. The order constraint is relaxed in proximity of the sequential pattern in order to discover the frequent itemsets that model its neighbourhood within data sequences. Furthermore, we have shown the relevance of LSR by considering the biomedical NER problem in which sequential patterns and sequential rules present limitations. LSR patterns offer a good trade-off between the high recall of sequential patterns and the high precision of sequential rules for this problem. Indeed, LSR patterns provide a surrounding context awareness that enables the disambiguation of the sequence, thanks to the analysis of its neighbourhood. Experiments, carried out on real datasets, show in these non-trivial cases, the power of our approach. Note that the use of LSR patterns for NER problems leads to an entirely automatic method in which extraction rules can be highly understood by a non-expert. Moreover, LSR patterns can be employed in other domains for the NER problem without effort since the method only considers sequences of tokens on its input.

There are several directions that can be followed to extend the ideas reported in this paper. Concerning the use of LSR in the NER problem, it would be interesting to consider richer input data. Instead of only considering sequences of tokens, we can introduce pieces of information as stemmas or part-of-speech analysis from computational linguistic. We are convinced that considering such pieces of information would result in an additional gain in both recall and precision when applying LSR patterns to the NER problem. It would also be interesting to use some LSR patterns as features in based-ML methods.

We argue that LSR patterns can also be used in many other contexts and problems. As an example, another use of LSR pattern could be the analysis of network datagrams in

the field of network security: an attack could be represented by (l, s, r) where l and r are the surrounding contexts before and after the attack. In this case, l is obviously more important than r in order to prevent the attack. It would also be interesting to apply LSR pattern mining to the discovery of interactions between genes and proteins so as to combine such knowledge with the one discovered in other types of data such as micro array datasets.

Acknowledgements

This work is partly supported by the ANR (French National Research Agency) funded project Bingo2 ANR-07-MDCO-014 (http://www.bingo2.greyc.fr). The work of Jiří Kléma was supported by Czech Ministry of Education under the Programme MSM 6840770012 Transdisciplinary Biomedical Engineering Research II. The Czech-French PHC Barrande project 'Heterogeneous data fusion for genomic and proteomic knowledge discovery' financed the travel expenses.

References

Agrawal, R. and Srikant, R. (1995) 'Mining sequential patterns', *Proc. of the 11th Int. Conf. on Data Engineering (ICDE'95)*, pp.3–14.

Cakmak, A. and Özsoyoglu, G. (2007) 'Annotating genes using textual patterns', in R.B. Altman, A.K. Dunker, L. Hunter, T. Murray and T.E. Klein (Eds.): *Pacific Symposium on Biocomputing*, World Scientific, pp.221–232.

Califf, M.E. and Mooney, R.J. (1999) 'Relational learning of pattern-match rules for information extraction', *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, AAAI, pp.328–334.

Chang, C-H., Kayed, M., Ramzy, M. and Shaalan, K.F. (2006) 'A survey of web information extraction systems', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 10, pp.1411–1428.

Charnois, T., Durand, N. and Kléma, J. (2006) 'Automated information extraction from gene summaries', *Proceedings of the ECML/PKDD Workshop on Data and Text Mining for Integrative Biology*, Berlin, Germany, pp.4–15.

Cohen, A.M. and Hersh, W.R. (2005) 'A survey of current work in biomedical text mining', *Brief Bioinform*, Vol. 6, No. 1, pp.57–71.

Cohen, B.K. and Hunter, L. (2004) 'Natural language processing and systems biology', *Artificial Intelligence Methods and Tools for Systems Biology*, Springer, pp.147–173.

De Raedt, L., Jager, M., Lee, S.D. and Mannila, H. (2002) 'A theory of inductive query answering', *Proceedings of the IEEE Conference on Data Mining (ICDM'02*), Maebashi, Japan, pp.123–130.

Dingare, S., Finkel, J., Nissim, M., Manning, C. and Alex, B. (2004) 'Exploring the boundaries: gene and protein identification in biomedical text', *Proceedings of the BioCreative (Critical Assessment of Information Extraction Systems in Biology) Workshop 2004*, Granada, Spain.

Fukuda, K., Tamura, A., Tsunoda, T. and Takagi, T. (1998) 'Toward information extraction: identifying protein names from biological papers', *Pacific Symposium Biocomputing (PSB'98)*, Hawaii, pp.362–373.

Garofalakis, M., Rastogi, R. and Shim, K. (1999) 'Spirit: sequential pattern mining with regular expression constraints', *Proc. of the 25th Int. Conf. on Very Large Databases (VLD'99)*, pp.223–234.

Humphreys, K., Demetriou, G. and Gaizauskas, R. (2000) 'Two applications of information extraction to biological science journal articles: enzyme interactions and protein structures', *Pacific Symposium Biocomputing*, Hawaii, pp.505–516.

Kim, J-H., Mithell, A., Attwood, T.K. and Hilario, M. (2007) 'Learning to extract relations for protein annotation', *BioInformatics*, Vol. 23, pp.253–257.

Kinoshita, S., Cohen, K.B., Ogren, P.V. and Hunter, L. (2005) 'Biocreative task 1a: entity identification with a stochastic tagger', *BMC Bioinformatics*, Vol. 6, Supp. 1.

Lee, D.S. and De Raedt, L. (2004) 'An efficient algorithm for mining string databases under constraints', *Knowledge Discovery in Inductive Databases 3rd Int. Workshop KDID'04*, Revised selected and invited papers, Springer-Verlag LNCS 3377, pp.108–129.

Leleu, M., Rigotti, C., Boulicaut, J-F. and Euvrard, G. (2003) 'Constraint-based mining of sequential patterns over datasets with consecutive repetitions', in N. Lavrac, D. Gamberger, H. Blockeel and L. Todorovski (Eds.): *PKDD*, Vol. 2838 of Lecture notes in Computer Science, pp.303–314, Springer.

Leser, U. and Hakenberg, J. (2005) 'What makes a gene name? Named entity recognition in the biomedical literature', *Briefings in Bioinformatics*, Vol. 6, No. 4, pp.357–369.

Mannila, H., Toivonen, H. and Verkamo, A. (1997) 'Discovery of frequent episodes in event sequences', *Data Mining and Knowledge Discovery*, Vol. 1, No. 3, pp.259–298.

Meger, N. and Rigotti, C. (2004) 'Constraint-based mining of episode rules and optimal window sizes', *Proc. of the 8th European Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD'04)*, Springer-Verlag, LNAI 3202, pp.313–324.

Nanni, M. and Rigotti, C. (2007) 'Extracting trees of quantitative serial episodes', *Knowledge Discovery in Inductive Databases 5th Int. Workshop KDID''06*, Revised selected and invited papers, Springer-Verlag, LNCS 4747, pp.170–188.

Nédellec, C., Bessieres, P., Bossy, R., Kotoujansky, A. and Manine, A-P. (2006) 'Annotation guidelines for machine learning-based named entity recognition in microbiology', *Proceedings of the Data and Text Mining in Integrative Biology Workshop, ECML/PKDD*, Berlin, pp.40–54.

Ng, R.T., Lakshmanan, V.S., Han, J. and Pang, A. (1998) 'Exploratory mining and pruning optimizations of constrained associations rules', *Proceedings of ACM SIGMOD'98*, ACM Press, pp.13–24.

Pei, J., Han, B., Mortazavi-Asl, B. and Pinto, H. (2001) 'Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth', *Proc. of the 17th Int. Conf. on Data Engineering (ICDE'01)*, pp.215–224.

Pei, J., Han, J. and Wang, W. (2002) 'Mining sequential patterns with constraints in large databases', *CIKM*, ACM, pp.18–25.

Pinto, H., Han, J., Pei, J., Wang, K., Chen, Q. and Dayal, U. (2001) 'Multi-dimensional sequential pattern mining', *CIKM*, ACM, pp.81–88.

Schmid, H. (1994) 'Probabilistic part-of-speech tagging using decision trees', *International. Conference on New Methods in Language Processing*.

Smith, L., Tanabe, L., Ando, R., Kuo, C., Chung, I., Hsu, C., Lin, Y., Klinger, R., Friedrich, C., Ganchev, K., Torii, M., Liu, H., Haddow, B., Struble, C., Povinelli, R., Vlachos, A., Baumgartner, W., Hunter, L., Carpenter, B., Tsai, R., Dai, H., Liu, F., Chen, Y., Sun, C., Katrenko, S., Adriaans, P., Blaschke, C., Torres, R., Neves, M., Nakov, P., Divoli, A., Maña López, M., Mata, J. and Wilbur, W. (2008) 'Overview of biocreative ii gene mention recognition', *Genome Biology*, Suppl. 2, Vol. S2, No. 9.

Soulet, A. and Crémilleux, B. (2005) 'An efficient framework for mining flexible constraints', in H.T. Bao, D. Cheung and H. Liu (Eds.): *Proceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'05)*, Hanoi, Vietnam, Springer, Vol. 3518 of LNAI, pp.661–671.

Srikant, R. and Agrawal, R. (1996) 'Mining sequential patterns: generalizations and performance improvements', in P.M.G. Apers, M. Bouzeghoub and G. Gardarin (Eds.), *EDBT*, Vol. 1057 of Lecture notes in Computer Science, pp.3–17, Springer.

Stefanowski, J. and Ziembinski, R. (2005) 'Mining context based sequential patterns', in P.S. Szczepaniak, J. Kacprzyk and A. Niewiadomski (Eds.): *AWIC*, Vol. 3528 of Lecture notes in Computer Science, pp.401–407, Springer.

Tanabe, L., Xie, N., Thom, L., Matten, W. and Wilbur, J. (2005) 'GENETAG: a tagged corpus for gene/protein named entity recognition', *BMC Bioinformatics*, Vol. 6, p.10.

Ukkonen, E. (1995) 'On-line construction of suffix trees', *Algorithmica*, Vol. 14, No. 3, pp.249–260.

Van Rijsbergen, C.J. (1979) *Information Retrieval*, 2nd ed., Dept. of Computer Science, University of Glasgow.

Yeh, A., Morgan, A., Colosimo, M. and Hirschman, L. (2005) 'BioCreAtIvE task 1A: gene mention finding evaluation', *BMC Bioinformatics*, Vol. 6, p.10.

Zaki, M. (2000) 'Sequence mining in categorical domains: incorporating constraints', *Proc. of the 9th Int. Conf. on Information and Knowledge Management (CIKM'00)*, pp.422–429.

Gene Interaction Extraction from Biomedical Texts by Sentence Skeletonization

Přemysl Vítovec
vitovpre@fel.cvut.cz
Jiří Kléma
klema@labe.felk.cvut.cz

Czech Technical University in Prague, Faculty of Electrical Engineering,
Department of Cybernetics

Abstract. The presented paper describes a method of text preprocessing improving the performance of sequential data mining applied in the task of gene interaction extraction from biomedical texts. The need of text preprocessing rises primarily from the fact, that the language encoded by any general word sequence is mostly not sequential. The method involves a number of heuristic language transformations, all together converting sentences into forms with higher degree of sequentiality. The core idea of enhancing sentence sequentiality results from the observation that the components constituting the semantical and grammatical content of sentences are not equally relevant for extracting a highly specific type of information. Experiments employing a simple sequential algorithm confirmed the usability of the proposed text preprocessing in the gene interaction extraction task. Furthermore, limitations identified during the result analysis may be regarded as guidelines for further work exploring the capabilities of the sequential data mining applied on linguistically preprocessed texts.

Keywords: gene interaction extraction, relation mining, text mining

1 Introduction

Gene interaction extraction from textual language representation can succeed only if language is understood correctly. In general, language comprehension proceeds through interpretation of grammar, semantics and pragmatics; omission of any of these components may cause the communication to fail. Individual language variants may differ in complexity of these components; biomedical language proves to be complex in all of them. Being the complexity extremely hard, any engineering approach has to omit some aspects by making assumptions, permitting relaxations etc. In case of sequential approach, which is focused in this project, this is expressed by assumption that language is of sequential nature. To diminish the negative effect of such a simplification while keeping the full potential power and flexibility of the sequential approach unchanged, text preprocessing needs to be employed. The text preprocessing method (*sentence skeletonization*) discussed in this paper builds on a priori linguistic knowledge.

2 Related Work

The methods commonly applied in the gene interaction extraction task include computational-linguistics based methods (mainly *language parsing*), *rule-based methods* and *machine learning based methods* [26].

Shallow parsing provides only partial decomposition of the sentence structure: part-of-speech tagged words are grouped into non-overlapping chunks of grammatically related words, whose relations are subsequently analyzed [10, 26]. Pustejovsky et al. [18] and Leroy et al. [13] accomplish the analysis using finite state automata. *Deep parsing*, in contrast, considers the entire sentence structure. Ahmed et al. [1] analyze the full parse by assigning predefined syntactic frames to parsed clauses, Skounakis et. al [23] automate the analysis by employing hidden Markov models (empiricist approach [26]). *Rule-based approaches* employ textual rules or patterns encoding relationships between entities [26]. Manually defined rules have been applied e.g. by Blaschke and Valencia [4] or Proux et al. [20]; systems capable of inducing rules automatically have been proposed e.g. by Huang et al. [11] or Hakenberg et al. [9] On the field of *machine learning based methods*, Kraven and Kumlien [7] employ bayesian classifier, Stapley and Benoit [24] use co-occurrence statistics, Airola et al. [2] extend the general graph kernel method introduced by Bunescu and Mooney [6] and construct a custom kernel to be passed to support vector machines.

Instead of being mutually exclusive, the three above principles rather supplement each other, as they each describe a different methodological aspect: parsing focuses on understanding the internal domain *structure*, rules on *encoding* internal dependencies and machine learning on *procedures* of revealing such dependencies. Advanced sequential approaches, like the *episode rules* proposed by Plantevit at al. [17], *encode* in fact the findings of machine learning based *procedures*. The missing *structural* view may be added by employing a reasonable text preprocessing.

The method of the text preprocessing discussed in this paper builds on the work of Miwa et al. [15], Jonnalagadda et al. [12] and Siddhartan [22], who propose various techniques transforming sentences into syntactically simpler structures.

3 Method Description

Text preprocessing discussed here converts a sentence into a set of structurally simpler word sequences called *skeletons*. Each skeleton *estimates* a subset of core semantical and syntactical features of the original sequence. Moreover, the language behind the skeleton is more reliably mirrored by the corresponding word sequence than in case of the original sentence. Thus, the *sequentiality* of skeletons is higher than the *sequentiality* of the original sentence, which makes them more suitable for applying sequential approaches. As a result of estimation, the skeleton set can not be regarded as a decomposition of the original sentence.

Skeletons are constructed following the bottom-up principle: being grounded at the *clause level*, they are further modified at the *sentence level*. The skeleton construction rely mostly on metalingual categories assigned to text by TreeTagger [21].

3.1 Clause Level

Problem Identification

(1)

G
the G gene
the G gene expression
the G gene expression in the cell
the activation of the G gene expression in the cell
the activation of the G gene expression in the eucaryotic cell

The Example 1 demostrates that altering a simple phrase by adequate language components causes the phrase to grow both to the left and to the right. Although there are limitations of such growth given by the demand of understandability, the space of all possible forms of phrases remains infinite. Assuming any semantically relevant sentence, e.g. *g inhibits X*, all the above phrases constitute a lexical paradigm for variable nominal argument *X*. This phenomenon will be referred to as *paradigmatic phrase space complexity*.

(2) *the [G1 activates G2]*

(3) *the expression of [G1 activates G2]*

In Example 2 *G1* binds to predicate *activates* (i.e. to the right), whereas in Example 3 *G1* binds to verbal noun *expression* (i.e. to the left). Therefore, in both sentences the marked subsequence represents different syntagma. This phenomenon will be referred to as *syntagmatic phrase space complexity*.

In conclusion, due to arbitrary phrase space complexity, the *positional distance* in the word sequence does not imply the underlying *language distance*.

Building Principles To deal with the above difficulties, the following principles have been defined as building blocks for the *clause level* text preprocessing:

Phrase structure reduction. The language sentence may be considered as a projection of a multidimensional, non-sequential language structure into a sequence of lexical elements. Backward mapping (i.e. word sequence interpretation) may be extremely difficult without fully qualified language knowledge. However, playing with paradigmatic relations (Example 1) reveals, that *semantically related* structures of different structural complexity can be placed at the same position, i.e. complex structures may be replaced with simpler ones without significant information loss. Applying recursively such transformations results in

clause level *skeleton*, which is assumed to hold or at least represent the core of the original clauses.

Operation atomicity. Working with the sentence as a whole implies facing the potential complexity of a general sentence. This can be avoided by operating on the lowest syntactical level: simplifying transformations considering only the closest context rely on *what we almost certainly know about the local language*. Moreover, atomicity and linguistic relevancy allow for heuristic qualifying and quantifying the additive semantic shifts caused by these transformations. However, the semantic shifts may be negligible, as in Example 4, where only attributes and appositional adjuncts are removed.

(4) $gene_{(att)}$ *G in eukaryotic*$_{(att)}$ *cells* $\rightarrow$ *G [in cells]*$_{(adj)}$ $\rightarrow$ *G*

Gene name propagation. Simplifying a word sequence can not proceed without removing words considered irrelevant. Language relevancy of words is closely related to their position in the phrase: word in *head* position holds the core meaning of the phrase and represents the minimal member of the corresponding paradigm, words at other position are linguistically less relevant. However, the language relevancy may conflict with the relevancy rising from the gene interaction extraction task, since gene entity names may occupy also attributive or adjunct positions. Therefore, to prevent the gene entity names from being removed, they need to be *propagated* to more stable positions. However, this procedure causes non-negligible (though measurable) shift in the semantic space of the given sentence (Example 5).

(5) $G_{(att)}$ *expression* $\rightarrow$ *G; expression of* $G_{(adj)}$ $\rightarrow$ *G*

Proximity assumption. Due to declared operation atomicity, the word sequence is never seen as a whole, but always locally. As a result, especially conjunction words may be ambiguous: being given only the immediate neighborhood, it may by hard to determine, what subsequences of the sentence actually constitute the arguments of the conjunction word. However, in case that both left and right neighboring words are of the same or related class, the following principle is applied: unless there is special reason for not treating them as arguments of the conjunction word (Example 7), they are treated as such (Example 6).

(6) *G1 activates [G2 and G3]* $\rightarrow$ *G1 activates G2+G3*

(7) *... expression [of G1] and [G2 activates]...*

The clause level transformations designed according to the above principles are summarized in Table 1.

Skeleton Construction The process of finding the clause skeletons can be roughly summarized into four steps: (1) reduce *noun chunks* into *minimal chunks* using the *left removal* and *forward propagation*; resolve *appositions* and *coordinations*, which results to a *nominal skeleton*. The remaining two steps are

Table 1. Clause level transformations. Legend: NC ∼ applicable within noun chunk; VC ∼ applicable within verb chunk; NCS ∼ applicable to noun chunk sequences.

Transformation	Cost	Type	Description
Leftřemoval(LR)	∼ 0	NC	Attribute removed, head preserved: *cell gene → gene; gene G → G*
Forward propagation (FP)	> 0	NC	Attribute moved to head position: *G expression → G*
Verb reduction	∼ 0	VC	Left verb form removed: *has activated → activated; is able to activate → activate*
Apposition reduction	∼ 0	NC	Concatenation + LR and FP *gene, G, → G; G1, G2, → G1+G2*
Coordination reduction	∼ 0	NC	Coordination + LR and FP *gene and G → G; gene and protein → protein; G1 and G2 → G1+G2*
Right removal	∼ 0	NCS	Appositional adjunct removed: *gene in cells → gene; G in cells → G*
Backward propagation	> 0	NCS	Appositional adjunct moved to preceding head: *expression if G → G*

specific to *verb skeletons*: (3) resolve nominal structures, mainly using the *right removal* and *backward propagation*; (4) resolve *appositions* and *coordinations* more freely. Following the path of abstraction, the above four steps may be further summarized in two steps: (I) investigate in details the internal structure of *noun chunk sequences*; (II) reduce the *noun chunk sequences* (if possible) to such forms which can be passed as arguments to clause verb predicate.

(8) *expression of G1 gene activates G2 induced protein G3 in mouse cells*
→ *expression of G activates G2 induced G3* (nominal skeleton)
→ *G1 activates G3* (verb skeleton)

Nominal structures are resolved and passed as arguments to clause predicates, i.e. nominal structures are subordinated to verb predicates. However, subset of nouns and adjectives may be also employed as predicates, i.e. they bind arguments: nouns, *gene entity words* or other nominal predicates. Nominal structures built around nominal predicates are saved in nominal skeletons before they are dissolved to become verb arguments. However, if they appear as arguments of a nominal predicate, they need to be stored in another nominal skeleton, before they are dissolved to become arguments of the superior nominal predicates. The procedure dealing with nested nominal predicates is not covered here due to limited space.

3.2 Sentence Level

Problem Identification

(9) *... it activates G2; ... and activates G2*

Even though the sentence stubs 9 seem incomplete with respect to their subjects, none of them has actually empty subject argument: both pronoun and unstated subject are valid syntactical subjects. However, these elements do not hold their own semantics; they only point to another language elements, thus propagating the once declared content to another sentence locations. The propagation naturally implies the *binding ability*: elements one representing the *holder* of the semantics and one the *pointer* (either explicit, or implicit) are clearly related to each other. This phenomenon will be referred to as *the existence of language pointers*.

(10) $[G1\ activating_{nominal}\ G2]\ interacts_{finite}\ with\ G3$

The predicative power of verb allows it to operate as top level node which divides clause in two regions containing (mainly nominal) arguments of the given verb. However, nominal verb forms (past participles, *ing*-forms) occur also within these regions (Example 10), while still preserving the verb syntactic behaviour. Moreover, some nominal verb structures tend to constitute their own subordinated clauses. An error in determining, which verb holds the role of sentence predicate, may lead towards loss of the sentence integrity. This phenomenon will be referred to as *existence of nominal verb forms*.

Building Principles The skeletons grounded at the clause level are further modified at the sentence level according to the following principles:

Mapping language pointers to corresponding values. Pointers need to be replaced by the elements they are pointing to, in order to prevent sequential algorithm from missing relation the element is involved in through this pointer. Correct mapping requires deep knowledge of discourse and information structure of general English sentence. Currently, the mapping employs only simple heuristic rules.

Mapping nominal verb forms to potential interaction predicates. To preserve the sentence semantical integrity, nominal verb forms are mapped to potential interaction predicates: verbs, nouns or adjectives with respect to current local context. The mapping follows complex heuristic rules extracted manually from random subsets of biomedical abstracts.

Assumption of neutral thematic structure. Scientific texts are assumed to follow the neutral textual principle: an entity is referred to not until it has been introduced. Therefore, only pointers pointing to the left are taken into account.

Operation minimality. In contrast to the clause level, transformations at the sentence level can not be evaluated using a reliable language based measure, since the context which needs to be covered is too large and therefore too versatile. To minimize the probability of making errors, only a minimum number of steps

are applied. Therefore, only those mappings are carried out, which cause any predicate to get two arguments, each containing at least one gene entity name.

The sentence level transformations designed according to the above principles are summarized in Table 2.

Table 2. Sentence level transformations. Legend: N ∼ within noun chunks, C ∼ within single clause, CC ∼ in context of two coordinate clauses; CS ∼ in context of clause and its subordinated clause.

Transf.	Appl.	Description
Explicit pointer mapping	N, CC, CS	Personal and possessive pronouns are mapped to gene entity names *... G1 consists of three exons and [it → G1] activates...* *... G1 and [its → G1] activation...*
Implicit pointer mapping	CC, C(S)	Unstated subjects are mapped to gene entity names *... G1 activates G2 and [¡none¿ → G1] associates...* *... G1 activates G2 by [¡none¿ → G1] associating...*
Ing-forms mapping	N, C(S)	Mapping *ing*-forms to nouns, adjectives or verbs
Participle mapping	N, C(S)	Mapping past participles to verbs or adjectives

4 Experiments

4.1 Testing Method

A simple sequential approach has been used to evaluate the effect of sentence skeletonization (i.e. improvement of sentence sequentiality) in the gene interaction extraction task: manually created, grammatically relevant patterns representing predication between two gene entities are matched against sentence skeletons, matching subsequences of sentence skeletons are considered to express interactions between the involved gene entities. Two features of this approach are essential:

(I) *Syntagmatic rigidity*: As the resulting sequentiality is the actual target of testing, the reference basis (i.e. what is certainly of sequential nature) represented here by the predefined sequential patterns should mirror the sequential principle in the clearest possible form in order to provide the most informative evaluation. Therefore, the time span between each two subsequent elements of all sequential patterns are set to one, i.e. neighboring tokens of a pattern have neighboring counterparts in the sentence skeleton, no time relaxation is allowed.

(II) *Paradigmatic latitude*: Instead of lexical elements, the sequential patterns are built (almost) exclusively from metalingual components, thus focusing on grammar rather than on the actual semantics (grammar is often a fundamental prerequisite for semantic integrity). The elements of sequential patterns

result from double abstraction: e.g. *noun*-token (i.e. second-level abstraction) of a sequential pattern covers four noun categories (i.e. singular, plural, proper etc.; first level abstraction) actually assigned to any English noun word by TREETAGGER [21]; i.e. any noun may be substituted for the *noun*-token.

The set sequential patterns consists of 29 patterns, 23 with a *verb predicate*, 3 with a *noun predicate* and 3 with an *adjective predicate*, e.g.: *gene verb gene; gene noun preposition gene; gene adjective gene.*

4.2 Experimental Data

The resulting sequentialty was evaluated on six biomedical corpora annotated both for gene entites and gene interactions: AIMED [16], CHRISTINE BRUN CORPUS [5], HPRD50 [14], IEPA [3], LLL05 [25] and BC-PPI [8]. All six corpora were handled in the same way according to the following four principles: (I) sentences are stemmed and assigned grammar tags using TREETAGGER [21]; (II) interactions employing more than two gene entities are converted into corresponding number of binary interactions (e.g. one ternary interaction corresponds to three binary interactions); (III) interacting gene pair, being detected in a corpus sentence, is counted only ones into performance measures (precision, recall, F-measure) regardless of how many times it is actually expressed in the sentence; (IV) a triple of two interacting genes and a binding *predicate* is counted only ones in the pattern analysis regardless of how many times it actually appears in the sentence.

4.3 Results

The overall performance of the presented approach in terms of *precision*, *recall* and *F-measure* is given in Table 3.

Table 3. Precision, recall and F-measure for all testing corpora

	AIMed	Brun	Hprd50	IEPA	LLL05	BC-PPI
Precision	0.49	0.62	0.81	0.74	0.87	0.36
Recall	0.46	0.47	0.61	0.59	0.72	0.65
F-measure	0.48	0.54	0.69	0.65	0.79	0.46

False negatives result mostly from the insufficient sequentiality of skeletonized sentences. Two corpora, LLL05 (providing excellent results) and BC-PPI (providing poor results), were analyzed in detail to identify both (a) the structures not covered by the sentence skeletonization, and (b) the factors causing the skeletonization to fail to improve the sentence sequentiality. A classification of such phenomena is given in Table 4.

False positives result either from (a) shortcommings of the sentence skeletonization, or (b) shortcommings of the sequential algorithm. (a) Provided that

Table 4. Analysis of false negatives: unhandled structures, confusing factors

	Category	Explanation
1	Incorrect tagging	E.g. *G1 binds@noun to G2*
2	Distance too long	E.g. multiple nested clauses before interaction is completed
3	Front-end arguments	E.g. *in addition to G2, G1 interacts with G3*
4	Nested *ing*-forms	E.g. *... by activating G2 encoding G3*
5	Higher level non-verb coordinations	E.g. *G1 interacts [with G2] and [with G3]*
6	Unresolved pointers	E.g. *high concentration of G1 induces G2, but low concentration(!) activates G3*
7	Misleading interpunction	E.g. *G1 and G2, interact with G3*
8	Different language forms	E.g. *complex of G1 and G2; G1 and G2 interact [with each other]*

stylistical correctness is guaranteed, the sentence complexity rises together with the complexity of the idea held by this sentence; thus, reducing the sentence complexity naturally distorts the underlying idea. The *atomicity principle* declared at the clause level typically prevents the corresponding transformations from exceeding the allowed level of distortion. Unfortunately, the *minimality principle* declared at the sentence level instead of the *atomicity principle* does not guarantee the same level of control. As a result, the corresponding transformations appear as error contributors more frequently. Moreover, their negative effect is often multiplied by coordinations, which distribute the error to all coordination participants. (b) Errors of the testing algorithm rise mostly from the omission of semantics: not every word holding the position of an interaction predicate does trully descibe an interaction. The overall performance on various corpora (Table 3) depends strongly upon the frequency of such confusing predicate candidates.

The atomicity allows to define a language based distance measure for estimative quantifying the semantic shift: the quantified overall semantic deviation from the original word sequence could be understood as a confidence in the obtained result (skeleton). However, the atomicity is currently declared only at the clause level. Therefore, any distance measure designed for estimating the overall semantic deviation from the original text representation will necessarily mirror exclusively the effect of clause level transformations. Experiments designed to find the optimal maximum allowed deviation by setting non-zero cost for both forward and background propagations (Table 1) proved, that such a measure is not sufficiently informative.

Pyysalo, Airola et al. [19, 2] use very similar approach to evaluate extraction performance of several approaches on five corpora, four of which are used in the presented experiments: AIMed, HPRD50, IEPA and LLL05. A comparison of some of them with the method proposed in this report is given in Table 5. Obviously, the presented approach achieves comparable results, even

though it was targeted only to evaluate the effect of sentence skeletonization and was not seriously meant as a full featured system for gene interaction extraction.

Table 5. Performance comparison. Legend: Graph kernel ∼ SVM based approach [2], RelEx ∼ approach involving deep parsing [19], Skel. + seq. ∼ the presented approach.

		AIMed	HPRD50	IEPA	LLL05
P	Graph kernel	0.529	0.643	0.696	0.725
	RelEx	0.40	0.76	0.74	0.82
	Skel. + seq.	0.49	0.81	0.74	0.87
R	Graph kernel	0.618	0.658	0.827	0.872
	RelEx	0.50	0.64	0.61	0.72
	Skel. + seq.	0.46	0.61	0.59	0.72
F	Graph kernel	0.564	0.634	0.751	0.768
	RelEx	0.44	0.69	0.67	0.77
	Skel. + seq.	0.48	0.69	0.65	0.79

5 Conclusion and Further Work

Since natural language is not sequential, linguistic preprocessing for sequential data mining (not limited to biomedical literature) can be understood as improving sentence sequentiality.

Based on a detailed analysis of biomedical texts, language phenomena breaking the sentence sequentiality have been identified. To deal with these obstacles, heuristic transformations have been designed, all of which are employed to convert a sentence into a set of skeletons, structures with improved level of sequentiality.. Sentence skeleton may be regarded as simplified form of the original sentence or sentence approximation (both grammatical and semantical), thus not being fully equivalent with the original sentence.

The impact of the sentence skeletonization has been evaluated using an intentionally simple, clearly sequential algorithm. By applying this algorithm in the gene interaction extraction task on skeletonized sentences from various biomedical corpora, limitations of the sentence skeletonization have been identified. Furthermore, the usability of pattern mining from sentence skeletons have been confirmed, provided that further improvements in sentence skeletonization will be made and a more advanced sequential algorithm will be used.

Sentence skeletonization will be further improved by applying the atomicity principle also at the *sentence level* and *text level*: this can be achieved by identifying the information flow between pairs of patternalized, i.e. further skeletonized clauses or sentences. Such method should not only solve the mapping problems, but it might also be helpful in dealing with various issues strongly related to pragmatics.

Furthermore, *episode rules*, an advanced general sequential approach proposed by Plantevit et al. [17], will be applied to sentence skeletons in the

gene interaction extraction task. Both lexical and metalingual information should be employed as features to balance the generalization potential and semantical relevancy of extracted rules.

Acknowledgment

The work of Přemysl Vítovec was funded by the Grant Agency of the Czech Technical University in Prague, grant No. SGS11/126/OHK3/TT/13. The work of Jiří Kléma was funded by the Czech Ministry of Education in the framework of the research programme Transdisciplinary Research in the Area of Biomedical Engineering II, MSM 6840770012.

References

1. Syed Toufeeq Ahmed, Deepthi Chidambaram, Hasan Davulcu, and Chitta Baral. Intex: a syntactic role driven protein-protein interaction extractor for bio-medical text. In *ISMB '05: Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases*, pages 54–61, Morristown, NJ, USA, 2005. Association for Computational Linguistics.
2. Antti Airola, Sampo Pyysalo, Jari Björne, Tapio Pahikkala, Filip Ginter, and Tapio Salakoski. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC Bioinformatics*, 9(Suppl 11):S2, 2008.
3. Daniel Berleant. *IEPA Corpus.* University of Arkansas at Little Rock, http://class.ee.iastate.edu/berleant/s/IEPA.htm. Accessed March 2010.
4. Christian Blaschke and Alfonso Valencia. The Potential Use of SUISEKI as a Protein Interaction Discovery Tool. *Genome Informatics*, 12:123–134, 2001.
5. Christine Brun. *Christine Brun Corpus.* http://www.biocreative.org/accounts/login/?next=/resources/. Accessed March 2010.
6. Razvan C. Bunescu and Raymond J. Mooney. A shortest path dependency kernel for relation extraction. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 724–731, Morristown, NJ, USA, 2005. Association for Computational Linguistics.
7. Mark Craven and Johan Kumlien. Constructing Biological Knowledge Bases by Extracting Information from Text Sources. In *Proceedings of the 7th Interactions conference on intelligent systems for molecular biology*, pages 77–86, 1999.
8. Jörg Hakenberg. *BC-PPI Corpus.* Humboldt-Universität zu Berlin - Institut für Informatik, http://www2.informatik.hu-berlin.de/ hakenber/corpora/. Accessed March 2010.
9. Jörg Hakenberg, Conrad Plake, Loic Royer, Hendrik Strobelt, Ulf Leser, and Michael Schroeder. Gene mention normalization and interaction extraction with context models and sentence motifs. *Genome Biology*, 9(Suppl 2):S14, 2008.
10. James Hammerton, Miles Osborne, Susan Armstrong, and Walter Daelemans. Introduction to special issue on machine learning approaches to shallow parsing. *The Journal of Machine Learning Research*, 2:551–558, 2002.
11. Minlie Huang, Xiaoyan Zhu, Yu Hao, Donald G. Payan, Kunbin Qu, and Ming Li. Discovering patterns to extract protein–protein interactions from full texts. *Bioinformatics*, 20(18):3604–3612, 2004.

12. Siddhartha Jonnalagadda, Luis Tari, Jörg Hakenberg, Chitta Baral, and Graciela Gonzalez. Towards effective sentence simplification for automatic processing of biomedical text. *CoRR*, 2010.
13. Gondy Leroy, Hsinchun Chen, and Jesse D. Martinez. A shallow parser based on closed-class words to capture relations in biomedical text. *Journal of Biomedical Informatics*, 36(3):145–158, 2003.
14. Ludwig-Maximilians-Universität München, Lehr- und Forschungseinheit für Bioinformatik, Institut für Informatik, http://code.google.com/p/priseinsttechuwt/-source/browse/trunk/PRISE/src/java/DEEPERsource/DEEPERsource/source/-resource/hprd50.xml?spec=svn3&r=3. *HPRD50 Corpus*. Accessed March 2010.
15. Makoto Miwa, Rune Saetre, Yusuke Miyao, and Jun'ichi Tsujii. Entity-focused sentence simplification for relation extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 788–796, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
16. Raymond J. Mooney. *AiMed*. University of Texas at Austin, https://wiki.inf.ed.ac.uk/TFlex/AiMed. Accessed March 2010.
17. M. Plantevit, T. Charnois, J. Klema, C. Rigotti, and B. Cremilleux. Combining sequence and itemset mining to discover named entities in biomedical texts: A new type of pattern. *International Journal of Data Mining, Modelling and Management*, 1:119–148, 2009.
18. J. Pustejovsky, J. Castafio, J. Zhang, M. Kotecki, and B. Cochran. Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations. In *Proceedings of the Pacific symposium on biocomputing*, pages 362–373, 2002.
19. Sampo Pyysalo, Antti Airola, Juho Heimonen, Jari Björne, Filip Ginter, and Tapio Salakoski. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(Suppl 3):S6, 2008.
20. Claude Roux, Denys Proux, Francois Rechenmann, and Laurent Julliard. An Ontology Enrichment Method for a Pragmatic Information Extraction System gathering Data on Genetic Interactions. In *Proceedings of the eight International conference on intelligent systems for molecular biology*, pages 279–285. AAAI Press, 2000.
21. Helmut Schmid. *Treetagger*. Institute for Computational Linguistics of the University of Stuttgart, http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/. Accessed March 2010.
22. Advaith Siddharthan. Syntactic simplification and text cohesion. *Language and Computation*, 4:77–109, 2006.
23. Marios Skounakis, Mark Craven, and Soumya Ray. Hierarchical Hidden Markov Models for Information Extraction. In *IJCAI*, pages 427–433, 2003.
24. B. J. Stapley and G. Benoit. Biobibliometrics: information retrieval and visualization from co-occurrences of gene names in Medline abstracts. In *Processing of the Pacific symposium on biocomputing*, pages 529–540, 2000.
25. Unité Mathématique, Informatique et Génome, http://genome.jouy.inra.fr/texte/-LLLchallenge/. *LLL05 Corpus*. Accessed March 2010.
26. Deyu Zhou and Yulan He. Extracting interactions between proteins from the literature. *Journal of Biomedical Informatics*, 41:393–407, 2008.

Chapter 5

Applications and Reviews

Líbalová et al. Particle and Fibre Toxicology 2012, 9:1
http://www.particleandfibretoxicology.com/content/9/1/1

PARTICLE AND FIBRE TOXICOLOGY

RESEARCH Open Access

Global gene expression changes in human embryonic lung fibroblasts induced by organic extracts from respirable air particles

Helena Líbalová[1,2], Kateřina Uhlířová[1], Jiří Kléma[3], Miroslav Machala[4], Radim J Šrám[1], Miroslav Ciganek[4] and Jan Topinka[1*]

Abstract

Background: Recently, we used cell-free assays to demonstrate the toxic effects of complex mixtures of organic extracts from urban air particles (PM2.5) collected in four localities of the Czech Republic (Ostrava-Bartovice, Ostrava-Poruba, Karvina and Trebon) which differed in the extent and sources of air pollution. To obtain further insight into the biological mechanisms of action of the extractable organic matter (EOM) from ambient air particles, human embryonic lung fibroblasts (HEL12469) were treated with the same four EOMs to assess changes in the genome-wide expression profiles compared to DMSO treated controls.

Method: For this purpose, HEL cells were incubated with subtoxic EOM concentrations of 10, 30, and 60 μg EOM/ml for 24 hours and global gene expression changes were analyzed using human whole genome microarrays (Illumina). The expression of selected genes was verified by quantitative real-time PCR.

Results: Dose-dependent increases in the number of significantly deregulated transcripts as well as dose-response relationships in the levels of individual transcripts were observed. The transcriptomic data did not differ substantially between the localities, suggesting that the air pollution originating mainly from various sources may have similar biological effects. This was further confirmed by the analysis of deregulated pathways and by identification of the most contributing gene modulations. The number of significantly deregulated KEGG pathways, as identified by Goeman's global test, varied, depending on the locality, between 12 to 29. The Metabolism of xenobiotics by cytochrome P450 exhibited the strongest upregulation in all 4 localities and *CYP1B1* had a major contribution to the upregulation of this pathway. Other important deregulated pathways in all 4 localities were ABC transporters (involved in the translocation of exogenous and endogenous metabolites across membranes and DNA repair), the Wnt and TGF-β signaling pathways (associated particularly with tumor promotion and progression), Steroid hormone biosynthesis (involved in the endocrine-disrupting activity of chemicals), and Glycerolipid metabolism (pathways involving the lipids with a glycerol backbone including lipid signaling molecules).

Conclusion: The microarray data suggested a prominent role of activation of aryl hydrocarbon receptor-dependent gene expression.

Keywords: air pollution, complex mixtures, HEL cells, CYP1B1, AhR, gene expression profile

* Correspondence: jtopinka@biomed.cas.cz
[1]Department of Genetic Ecotoxicology, Institute of Experimental Medicine, Academy of Sciences of the Czech Republic, 142 20 Prague 4, Czech Republic
Full list of author information is available at the end of the article

Background

Considerable efforts have been made to clarify the adverse effects of environmental pollution on human health [1]. Respirable ambient air particulate matter with an aerodynamic diameter < 2.5 μm (PM2.5) is a complex mixture consisting of a large number of chemicals, many of which are toxic and/or carcinogenic [2]. The mixtures of organic compounds to which the general population is exposed are not completely characterized since complex chemical analysis is very difficult. Investigations into the biological effects of ambient air particulate matter have involved a number of different approaches, including the study of particle-induced genotoxicity. Although hundreds of genotoxic compounds have been identified in ambient air, less than 25 of these compounds are routinely monitored [3]. Therefore, a biological approach based on specific toxic effects, such as direct or indirect reactivity with DNA or mutagenicity of complex mixture components might represent a suitable alternative [4,5]. The toxic effects of ambient air particulate matter (PM) are most frequently associated with chemicals bound onto the surface of the PM and/or with the particles themselves [6,7]. Some studies suggest that the genotoxic effects of PM are induced by polycyclic hydrocarbons (PAHs) and their derivatives forming the organic fraction of PM [1,8,9]. Other studies indicate that some metals forming PM may catalyze the oxidative damage of DNA [10-12]. Much less attention has been paid to nongenotoxic mechanisms of the toxic effects of chemicals bound onto PM2.5, although complex mixtures of air pollutants are known to contain various tumor promoters [13,14]. It has been repeatedly demonstrated that some PAHs, such as benzo[a]pyrene (BaP), form DNA adducts, after their metabolic activation by cytochrome P450 enzymes [15-18]. However, the PAHs, which activate aryl hydrocarbon receptor (AhR), induce several AhR-dependent nongenotoxic effects associated with tumor promotion [19,20]. PAHs have been reported to contribute to antiapoptotic effect of PM via activation of AhR in human bronchial epithelial cells [21] and AhR-dependent induction of cell proliferation, another hallmark of tumor promotion, after exposure to the extract of reference airborne particles has been described in liver epithelial cells [14]. Moreover, another group of PAHs (fluoranthene, pyrene) is known to exhibit tumor promoting activity via inhibition of intercellular communication [13,22].

Several attempts have been made to study the toxic effects of both artificial and real mixtures of environmental air pollutants, including PAHs, in various cell cultures [23]. The recent progress of "omics" technology in toxicology has allowed more insight into the mechanisms of the toxic effects of complex mixtures [24]. This technology offers the ability to query the entire genome after exposure to a complex mixture of compounds, permitting characterization of the biological effects of such exposure and the mechanisms of action involved. Significant attention has been paid to the global gene expression changes caused by complex mixtures, such as cigarette smoke and its condensate, diesel exhaust and carbon black. However, only a few studies have dealt with ambient dust particles (reviewed in [24]). The genome-wide study, dealing with particles from urban dust (standardized SRM1649a) in a human cell line in vitro, indicated deregulation of genes involved in DNA repair, peroxisome proliferation, metabolism and changes in tissue growth factors and oncogenes [25]. In human aortic endothelial cells exposed to the ambient particular matter, the modulation of gene expression included upregulation of metabolism of xenobiotics and proinflammatory responses [26].

In this study, the toxicogenomic approach was used to identify genes and particularly the biological pathways involved in the action of mixtures of organic air pollutants adsorbed onto respirable air particles (PM2.5). As a model system, human embryonic lung fibroblasts (HEL) which have been repeatedly shown to be a suitable model for toxicity studies of individual compounds as well as artificial and environmental mixtures, were used [9,27]. Importantly, embryonic fibroblasts have distinctive differentiation status compared with other lung cell models and therefore, unique gene expression changes might be expected. Changes in the whole genome expression profiles induced by extractable organic matter (EOM) from the PM2.5 particles in HEL cells were analyzed at subtoxic EOM concentrations and significantly deregulated genes and biological processes were identified. Moreover, changes in gene expression profiles for various localities (differing by the sources and extent of air pollution) were compared with the aim of identifying exclusive changes in gene transcription profiles corresponding to the air pollution exposure.

Results

Air sampling

The occurrence of organic compounds in the air is dependent on their physical properties, there are present in the gas phase, partially or completely adsorbed to the particles present in the air. This fact complicates the procedures for air sampling and the interpretation of the observed concentrations or toxic effects. PAHs with two to three cycles are present in the air under normal physical conditions in the gas phase (partly but can also be adsorbed on air particles, e.g. fluoranthene), PAHs with four cycles (e.g., pyrene) are distributed both in the gas and particulate phase, and PAHs with five or more cycles (benzo[*a*]pyrene, benzofluoranthenes, dibenzoanthracenes and dibenzopyrenes) are almost entirely adsorbed on particles [28-30]. This study was also focused on the determination of organic extractable compounds bound

to the particulate matter in the air, which in terms of genotoxicity, carcinogenicity and dioxin-like toxicity represent the highest risk.

The basic characteristics of PM2.5 sampling such as GPS coordinates, volume of sampled air, concentrations of PM2.5 and EOM in all localities are summarized in Table 1. The highest air pollution level in terms of PM2.5 was found in the industrial area of Ostrava-Bartovice (1.5-fold and more than 3-fold higher than in Ostrava-Poruba and Trebon, respectively).

Chemical characterization of ambient air particulate matter

To evaluate the chemical characterization of ambient air particulate matter ($PM_{2.5}$), many classes of organic contaminants were analyzed (Additional file 1). The highest concentrations were found for n-alkanes (77.4 - 89.5 ng/m^3), ten U.S. EPA PAHs (parent PAHs prioritized by U.S. EPA, ranged from 5.89 to 76.3 ng/m^3), other PAHs (other parent compounds with significant toxicological and indicatory characteristics, 3.28 - 44.3 ng/m^3) and oxidized PAHs (2.02 - 36.1 ng/m^3) (Table 2). Assuming that traffic emitted n-alkanes and PAHs in similar proportions, then the approximately one order of magnitude higher PAH emissions in the hot spot site Bartovice is caused by emissions from other, mainly local industrial sources. Like n-alkanes, other contaminants associated with emissions from traffic (UCM, terpanes, triterpanes and steranes) were present in the samples. In addition to these compounds, sterols (mainly of plant origin, stigmasterol, β-sitosterol, β-amyrin, β-amyrin and lupeol), ubiquitous dialkyl-esthers of phthalic acid, which is still used as a softener for plastics based on polyvinyl chloride, and other industrial contaminants (bisphenol A, benzophenone, etc.) were also found. Different, more abundant individual U.S. EPA PAHs were found in the sites under study. Pyrene (a marker of pyrogenic sources of PAHs) dominated in Bartovice, Poruba and Trebon; Indeno[1,2,3-*cd*]pyrene (a marker of traffic sources) prevailed in the site Karvina (Table 3). The average concentration of benzo[*a*]pyrene ranged from 0.55 (Trebon site) to 5.98 ng/m^3 (hot spot site Bartovice). The difference in B[a]P was much higher in terms of B[a]P than in terms of PM2.5 (3-fold higher PM2.5 levels and more than 10-fold higher B[a]P levels in Ostrava-Bartovice and Trebon, respectively).

Gene expression changes induced by EOMs

Gene expression profiling using the Illumina microarray platform and pathway analysis was used to identify deregulated genes and biological processes in HEL cells following 24 h exposure to EOM from each locality at three subtoxic concentrations (10, 30, 60 μg EOM/ml). Gene expression levels were compared to control HEL cell cultures treated with DMSO only.

Deregulated transcripts and genes

We first identified differential gene expression in each EOM dose from all 4 localities. A full list of deregulated genes is available as Additional file 2. The number of deregulated transcripts with adjusted *P*-value < 0.05, average expression level (AvgExp) > 4, and $\log_2$ FC (fold change) > |1|exhibiting a positive dose response for all 4 localities is shown in Figure 1. More than 1200 transcripts were deregulated at the highest dose of 60 μg EOM/ml for the heavily polluted area of Ostrava-Bartovice, while after the exposure to the extract sample from Ostrava-Poruba (6 km from Ostrava-Bartovice) only about 700 genes were deregulated. Significant overlap of deregulated transcripts was observed between the localities (Figure 2). More than 360 transcripts were deregulated simultaneously in all 4 localities for EOM at the concentration of 60 μg EOM/ml. This number represented approximately 30% of all deregulated genes for Ostrava-Bartovice, 50% for Ostrava-Poruba, 68% for Karvina, and 36% for Trebon sample. Despite this significant overlap, 388 transcripts (32%) were exclusively deregulated in cells treated with EOM (60 μg/ml) from Ostrava-Bartovice, while only 58 (8%), 37 (7%), and 178 (18%) transcripts were deregulated by samples from Ostrava-Poruba, Karvina, and Trebon, respectively.

Table 1 Basic characteristics of PM2.5 sampling in various localities of the Czech Republic

Locality [GPS coordinates]	Sampling period	Air volume [m^3]	PM [μg/m^3]	EOM [μg/m^3]
Ostrava-Bartovice [49°48'07"N, 18°20'56"E]	1.3.-4.4. 09	29,900	36.7	13.0
Ostrava-Poruba [49°48'07"N, 18°20'56"E]	1.3.-31.3. 09	35,200	25.8	8.05
Karvina [49°48'07"N, 18°20'56"E]	1.4.-5.5. 09	47,400	n.a.*	9.16
Trebon [49°00'15"N, 14°45'56"E]	19.11.-17.12. 08	44,700	11.4	4.15

*Missing for technical reasons

Table 2 Concentration of contaminant groups in the extracts from PM2.5 (ng/m^3)

Contaminant classes	Contaminant groups	Ostrava-Bartovice	Ostrava-Poruba	Karvina	Trebon
Polycyclic aromatic compounds	U.S. EPA PAHs (10)*	76.3	13.6	17.1	5.89
	other PAHs (29)	44.3	7.14	9.29	3.28
	alkylated PAHs (46)	29.7	7.36	5.88	4.98
	oxidized PAHs (7)	36.1	11.0	12.8	2.02
	N-heterocyclic PAHs (PANHs) (13)	19.9	4.27	3.13	0.46
	S-heterocyclic PAHs (PASHs) (8)	9.99	2.27	1.80	0.42
	nitrated PAHs (15)	0.38	0.05	0.04	0.02
	dinitrated PAHs (3)	0.00097	0.00008	0.00005	0.00005
Hydrocarbon markers	n-alkanes (29)	84.4	89.5	77.4	89.2
	UCM (unresolved complex mixture) **	20.1	15.3	9.19	12.7
	terpanes (15)	14.8	9.34	5.64	4.14
	triterpanes (13)	4.49	3.16	1.49	3.73
	steranes (19)	2.24	2.68	0.85	1.76
Sterols	faecal sterols (8)	0.38	0.12	0.17	0.88
	phytosterols (5)	1.04	0.20	0.59	1.63
Industrial contaminants	musk compounds (9)	0.01	0.003	0.003	0.06
	dialkyl-phthalates (6)	0.75	0.60	0.34	2.08
	bisphenol A	0.18	0.09	0.10	0.09
	benzophenone	0.24	0.05	0.04	0.04
	isomyristate	0.19	0.10	0.07	0.46

* the numbers of quantified compounds are in parentheses

** mixture of thousands cyclic and branched saturated hydrocarbons forming a characteristic hump on the GC chromatogram of the fraction of non-polar compounds of air particulate matter

To further evaluate the differences in gene expression profiles between localities, principal component analysis (PCA) was performed (Figure 3). The data did not exhibit any significant clustering according to the locality, suggesting similarities in expression profiles. In contrast, clusters separating individual EOM concentrations were observed (ellipses in Figure 3 indicate 95% confidence interval). Further statistical analysis of the expression data was focused on the deregulated pathways involving the levels of all detectable transcripts, and not only significantly deregulated transcripts.

Deregulated pathways

To identify deregulated KEGG pathways for the individual localities, all 3 EOM concentrations (10, 30, and 60 μg/ml) were combined for the analysis. The main reason for that was to identify deregulated pathways in individual localities for the whole concentration range. The complete list of significantly deregulated pathways resulting from EOM-treated HEL 12469 cells, as identified by Goeman's global test, is shown in Table 4. Although the analysis on the level of individual transcripts identified an almost 2-fold higher number of deregulated genes for Ostrava-Bartovice than for Ostrava-Poruba (1212 vs. 719 for 60 μg EOM/ml), the number of deregulated pathways was higher for Ostrava-Poruba than for Ostrava-Bartovice (29 vs. 18). The pathway exhibiting the strongest deregulation in all 4 localities was the Metabolism of xenobiotics by cytochrome P450. The main genes contributing to the deregulation of this pathway included upregulation of *CYP1B1,*

Table 3 Selected priority U.S. EPA PAHs adsorbed on the PM2.5 collected in various localities (ng/m^3)

Compound name	Ostrava-Bartovice	Ostrava-Poruba	Karvina	Trebon
Fluoranthene	11.6	2.48	1.72	1.00
Pyrene	13.9	2.61	1.76	1.01
Benz[a]anthracene	11.6	1.62	1.67	0.50
Chrysene	9.06	1.96	2.19	0.89
Benzo[b]fluoranthene	3.89	0.55	1.03	0.31
Benzo[k]fluoranthene	4.34	0.69	1.18	0.32
Benzo[a]pyrene	5.98	1.31	2.26	0.55
Dibenz[a, h]anthracene	1.16	0.21	0.16	0.09
Benzo[ghi]perylene	5.15	0.80	1.83	0.51
Indeno[1,2,3-cd]pyrene	9.67	1.38	3.30	0.72

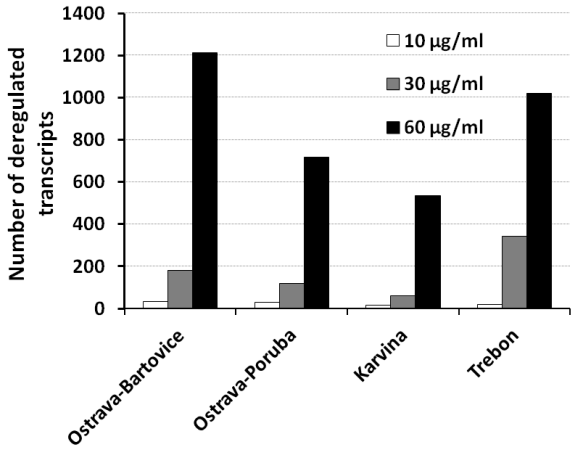


Figure 1 Number of significantly deregulated transcripts (adjusted *P*-value < 0.05, average expression level (AvgExp) > 4, and $\log_2$ FC (fold change) > |1|) in HEL 12469 cells treated with various subtoxic concentrations of EOMs from PM2.5 collected in four localities of the Czech Republic. List of all deregulated genes is available as Supplementary material.

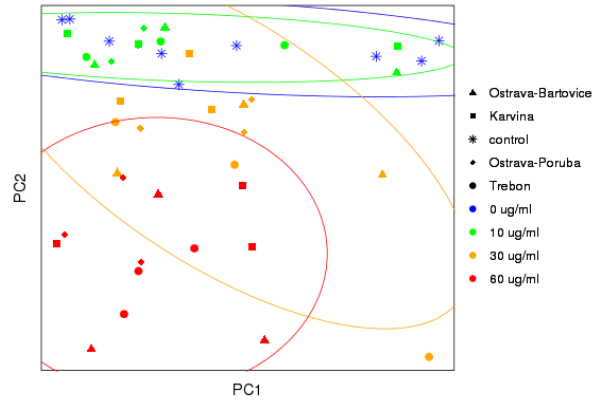


Figure 3 Principal component analysis of the normalized gene expression data from microarrays for all localities and EOM concentrations. Ellipses bound clusters of the various concentrations (95% confidence interval).

MGST1 (2 transcripts), *GSTM5*, and *GSTO1* (Figure 4). The significance of this contribution as well as the correlations between transcripts are shown in Figure 5. These results suggest a crucial role of *CYP1B1* upregulation and its correlation with the expression of other genes encoding detoxifying enzymes. All the genes depicted in Figure 4 were upregulated. Five other pathways were significantly deregulated in all 4 localities: Steroid hormone biosynthesis (driven mostly by *CYP1B1* and aldo-ketoreductases), ABC transporters, Wnt signaling pathway, TGF-β signaling pathway, and Glycerolipid metabolism. The genes with the highest contribution to the deregulation of these pathways at various localities are summarized in Figure 4. Together with the pathways deregulated in all localities, Figure 4 summarizes 5 other toxicologically important pathways (Drug metabolism by cytochrome P450, Glutathione metabolism, Gap junction, Arachidonic acid metabolism, p53 signaling) deregulated in at least 1 of the 4 localities. For each pathway, selected genes mainly contributing to pathway deregulation are shown.

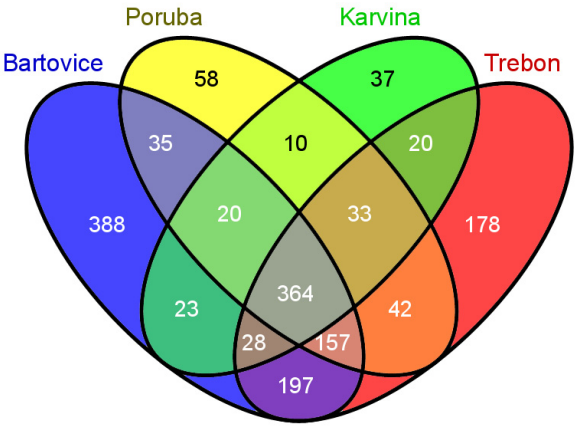


Figure 2 Venn diagram representing numbers of common and locality-specific deregulated genes following 24-h treatment of HEL cells with 60 µg EOM/ml, relative to DMSO used as a solvent control (adjusted *P*-value < 0.05, average expression level (AvgExp) > 4, and $\log_2$ FC (fold change) > |1|).

Quantitative real-time PCR verification

The gene expression of 11 selected significantly deregulated genes from microarray data in HEL cells was verified by qPCR. These genes include *CYP1B1, MGST1, NKD2, BMP2, SMAD3, TBXAS1, CCND2, PTGS2, TJP1, WNT2*, and *ID2*. The transcripts were selected to represent various deregulated pathways (Figure 4). Transcript levels of each selected gene were measured in each locality (Figure 6 A-D). In most cases, data proved dose-dependent up- or downregulation as indicated by Jonckheere-Terprsta monotonicity test. With the exception of the downregulation of *SMAD3* gene involved in *TGF*-β and Wnt signaling pathways, all other transcripts verified by qPCR were closely correlated (Figure 7). The mean correlation across all the transcripts and localities was r = 0.91, the mean correlation without *SMAD3* gene was r = 0.96 (r is the Pearson correlation coefficient).

Discussion

This study aimed to use human embryonic lung fibroblasts (HEL12469) as a model of target tissue for inhalation exposure, to identify biological processes and pathways involved in the toxic effects of organic extracts from respirable ambient air particles collected in 4 localities of the Czech Republic differing in the extent and

Table 4 Pathways significantly deregulated after EOM-treatment of HEL 12469 cells as identified by Goeman's global test

ID	KEGG pathway	Ostrava-Bartovice	Ostrava-Poruba	Karvina	Trebon
			Adj. p-value*		
980	**Metabolism of xenobiotics by cytochrome P450**	**4.06E-04**	**2.97E-05**	**4.71E-04**	**1.70E-03**
4270	Vascular smooth muscle contraction	**3.08E-04**	**2.06E-03**	*1.09E-01*	**2.41E-03**
4310	**Wnt signaling pathway**	**5.14E-03**	**1.32E-04**	**7.07E-03**	**2.88E-03**
30	Pentose phosphate pathway	**5.64E-03**	**8.84E-03**	*1.27E-01*	**3.19E-02**
140	**Steroid hormone biosynthesis**	**1.03E-02**	**2.58E-03**	**4.29E-03**	**3.37E-02**
2010	**ABC transporters**	**1.17E-02**	**1.86E-03**	**3.51E-02**	**4.34E-03**
561	**Glycerolipid metabolism**	**1.50E-02**	**2.63E-03**	**4.38E-02**	**5.03E-03**
770	Pantothenate and CoA biosynthesis	**1.64E-02**	**4.78E-02**	**3.68E-02**	*1.46E-01*
4540	Gap junction	**1.80E-02**	**3.50E-02**	*8.14E-02*	**3.63E-02**
4350	**TGF-beta signaling pathway**	**1.84E-02**	**3.07E-02**	**3.09E-02**	**1.29E-02**
4115	p53 signaling pathway	**2.39E-02**	*1.70E-01*	*2.74E-01*	7.77E-02
520	Amino sugar and nucleotide sugar metabolism	**2.58E-02**	**6.46E-03**	*1.22E-01*	*9.54E-02*
600	Sphingolipid metabolism	**2.63E-02**	*2.72E-01*	*8.92E-02*	**3.45E-02**
52	Galactose metabolism	**2.63E-02**	**1.97E-02**	*1.86E-01*	*7.77E-02*
620	Pyruvate metabolism	**2.89E-02**	**2.68E-02**	*1.00E+00*	**4.05E-02**
982	Drug metabolism - cytochrome P450	**3.01E-02**	**9.84E-05**	**1.49E-03**	*9.04E-02*
480	Glutathione metabolism	**3.08E-02**	2.13E-03	**7.91E-03**	*6.30E-02*
72	Synthesis and degradation of ketone bodies	**3.83E-02**	*2.10E-01*	*2.74E-01*	*1.00E+00*
4340	Hedgehog signaling pathway	*5.23E-02*	**2.22E-03**	**5.75E-02**	**4.33E-02**
5217	Basal cell carcinoma	*5.06E-02*	**5.04E-03**	**4.86E-02**	*5.16E-02*
40	Pentose and glucuronate interconversions	*6.10E-02*	**5.70E-03**	*1.59E-01*	*6.37E-02*
4142	Lysosome	*7.53E-02*	**5.78E-03**	*4.30E-01*	*7.05E-02*
565	Ether lipid metabolism	*5.47E-02*	**1.23E-02**	**1.52E-02**	**1.38E-02**
4614	Renin-angiotensin system	*5.43E-02*	**1.33E-02**	*1.22E-01*	*6.17E-01*
590	Arachidonic acid metabolism	*5.71E-02*	**1.97E-02**	*7.05E-02*	**4.70E-02**
4744	Phototransduction	*1.56E-01*	**2.07E-02**	*4.15E-01*	2.22E-02
511	Other glycan degradation	*1.09E-01*	**2.64E-02**	*5.96E-01*	*7.55E-02*
4610	Complement and coagulation cascades	*1.15E-01*	**2.85E-02**	*9.83E-01*	*4.56E-01*
5222	Small cell lung cancer	*1.48E-01*	**3.07E-02**	*4.63E-01*	*7.55E-02*
4612	Antigen processing and presentation	*4.26E-01*	**3.35E-02**	*5.52E-01*	*2.08E-01*
5140	Leishmaniasis	*1.18E-01*	**3.58E-02**	*2.20E-01*	*5.18E-02*
4145	Phagosome	*5.26E-02*	**4.38E-02**	*1.00E+00*	*1.16E-01*
564	Glycerophospholipid metabolism	*3.80E-01*	*1.08E-01*	*7.85E-01*	**6.10E-03**
4730	Long-term depression	*2.77E-01*	*4.64E-01*	*1.00E+00*	**7.49E-03**

KEGG pathways deregulated in all localities are in bold. PM2.5 samples were collected in 4 localities of the Czech Republic, and extractable organic matters (EOMs) were prepared as described in Materials and Methods.

*The procedure of Holm for control of the family-wise error rate [62].

sources of environmental pollution. For this purpose, cell cultures were treated with subtoxic concentrations of organic extracts from PM2.5 particles collected by high volume filter sampling. To ensure the complexity of the study, the whole genome RNA expression microarray covering 48 k of gene transcripts was used. The major findings of the study suggest that multiple genes involved in various biological pathways were deregulated in a dose-dependent manner, and the highest number of transcripts was deregulated in Ostrava-Bartovice, a residential part of Ostrava city which is mostly polluted by heavy industry, such as steel works and coke ovens located in the immediate vicinity [31]. Taking into account substantial differences in major air pollution sources among the localities, the high number of commonly deregulated genes seems to be surprising (30-68%, depending on the locality). This is further supported by the qualitative similarities in the chemical composition of the organic extracts from all 4 localities (determined more than 200 aromatic compounds), particularly by the results from the principal component analysis of gene expression profiles which indicated clustering according to the EOM concentration, but not according to the locality. However, this similarity was least for the most

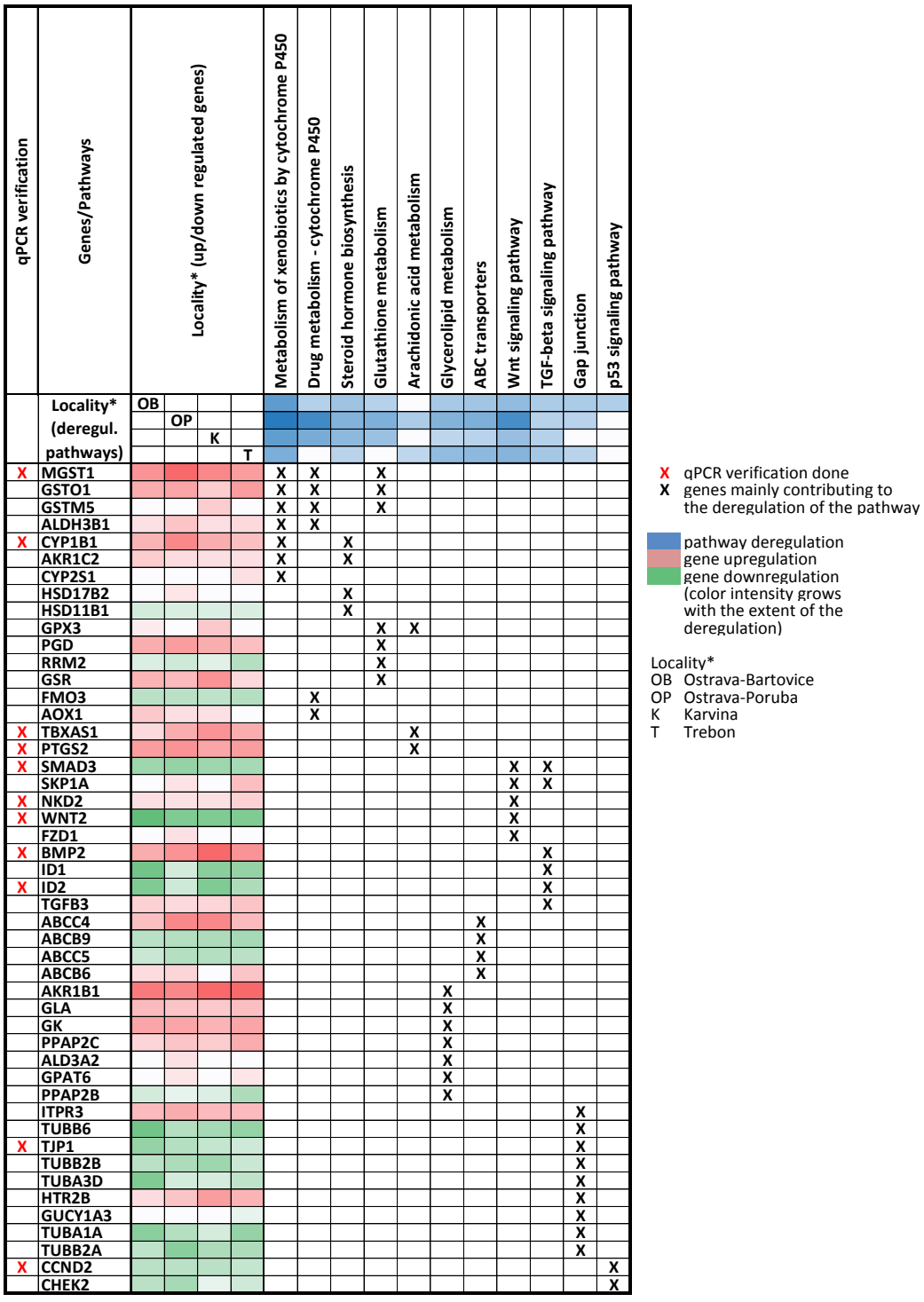


Figure 4 Selected deregulated pathways and genes mainly contributing to their deregulation in various sampling localities.

polluted area of Ostrava-Bartovice, where 32% of deregulated transcripts were exclusive to this locality, which was much more than for the 3 remaining areas. Chemical analysis revealed higher relative content of some of PAHs and nitrated PAH derivatives, which may at least explain a slightly different gene expression responses. However, the major modulations of gene expression were dependent on activation of aryl hydrocarbon receptor (AhR).

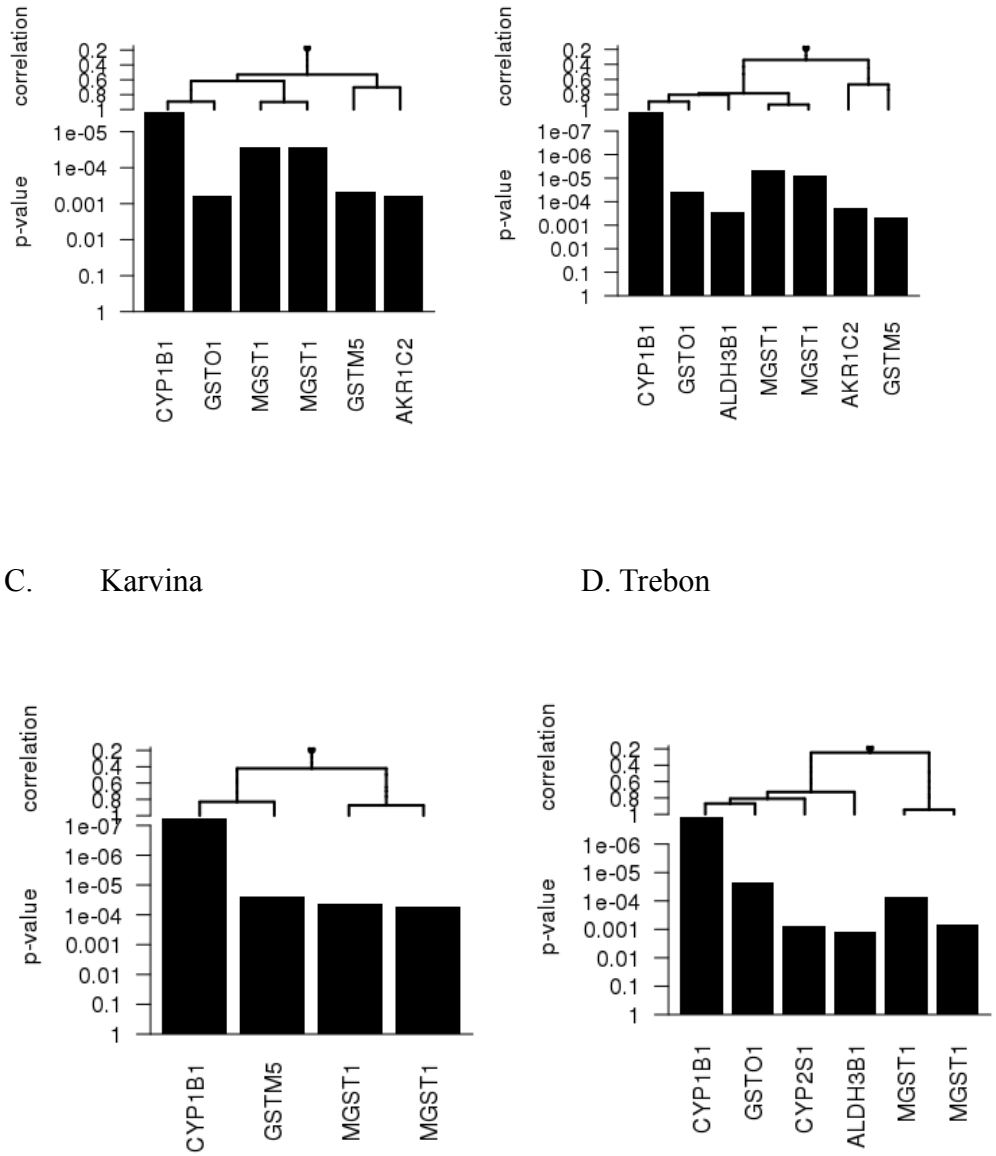


Figure 5 Genes mainly contributing to deregulation of the KEGG pathway, metabolism of xenobiotics by cytochrome P450, in various sampling localities. All depicted genes were upregulated.

For better biological interpretation of the observed gene expression changes, we identified deregulated pathways without preliminary discrimination of up- and downregulated genes. The data indicated that some pathways were significantly affected by both up- and downregulated genes. We were also interested in pathways that contained a large number of genes whose regulation was associated with the EOM concentration in "a small way" (still under the significance threshold for each individual gene). Therefore, the preliminary discrimination of up- and downregulated genes may result in loss of some deregulated pathways. Furthermore, for the purpose of pathway analysis, we did not discriminate between individual EOM concentrations used in the treatment of HEL cells.

Taking into account the high levels of PAHs, their derivatives and many other compounds bound to PM2.5

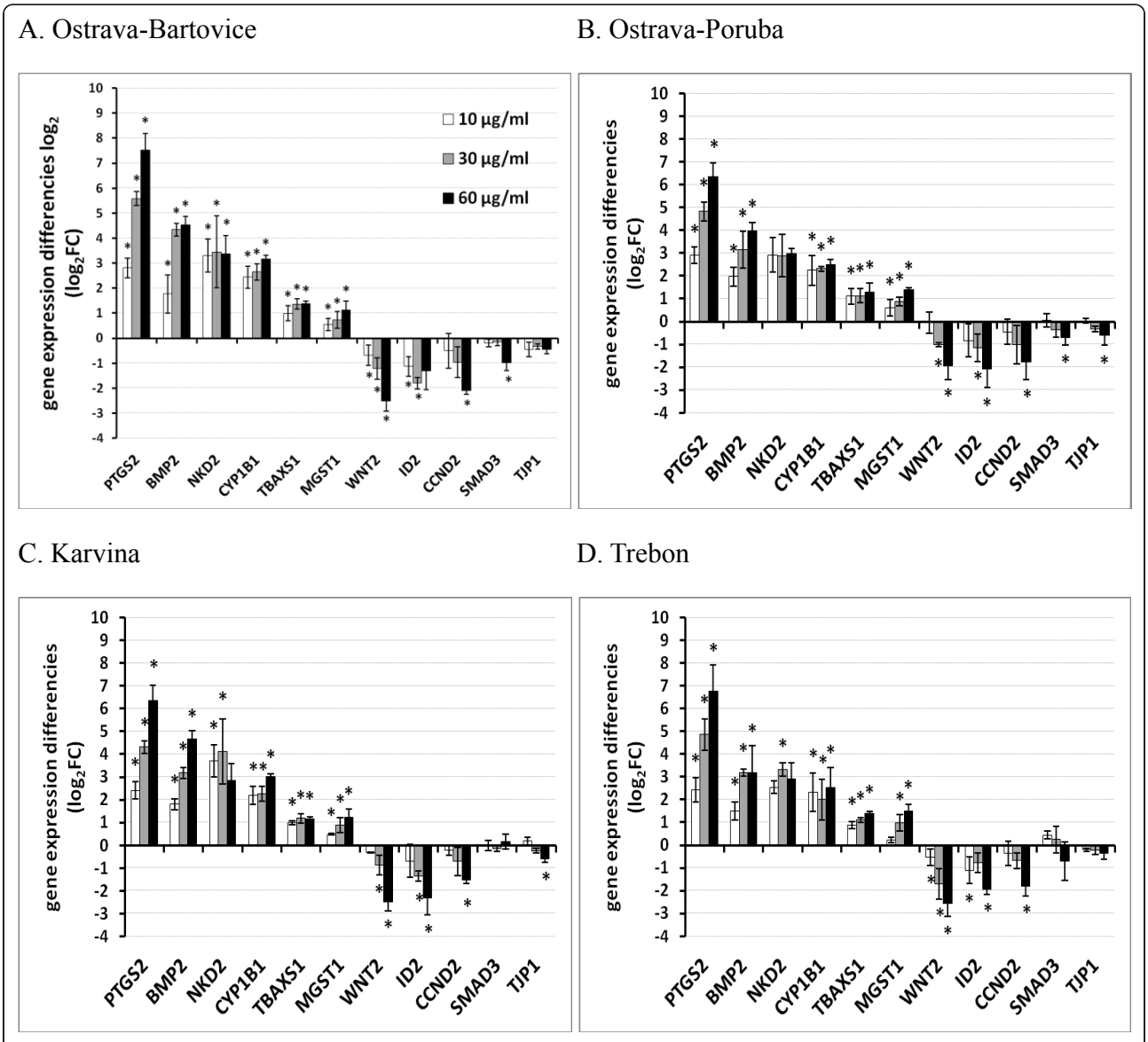


Figure 6 Gene expression changes (gene expression difference log_2) of 11 selected genes representing various deregulated pathways (see Table 4) as detected by quantitative RT-PCR. HEL cells were treated with 10, 30 and 60 μg/ml of organic extracts from PM2.5 particles. All transcripts from all localities (with the exception of SMAD 3 for Karvina and Trebon and TJP1 for Trebon) were significantly deregulated in a dose-dependent manner (*P*-value of the Jonckheere-Terpstra monotonicity test was < 0.05).

[2], it is not too surprising that the strongest deregulation in this study was observed for the KEGG pathway, Metabolism of xenobiotics by cytochrome P450. Among the many upregulated metabolic enzymes in this pathway, *CYP1B1* dominated in all localities. *CYP1B1* is a mixed-function monooxygenase, which metabolizes mainly polycyclic aromatic hydrocarbons, N-heterocyclic amines, arylamines, aminoazodyes and several other carcinogens [32]. Besides its role in the metabolism of xenobiotics, *CYP1B1* is also involved in the metabolism of cholesterol, steroid hormones, arachidonic acid and other lipids, metabolism of retinoic acid as well as in vitamin D3 synthesis and metabolism [33].

The induction of *CYP1B1* by complex mixtures such as tobacco smoke or airborne particles has been observed [14,25,34]. It is well known that the *CYP1B1* gene is under the regulatory control of the AhR and many PAHs are known to induce *CYP1B1* and their own metabolism through binding to and activation of the AhR [35]. The AhR is a ligand-activated transcription factor which has a

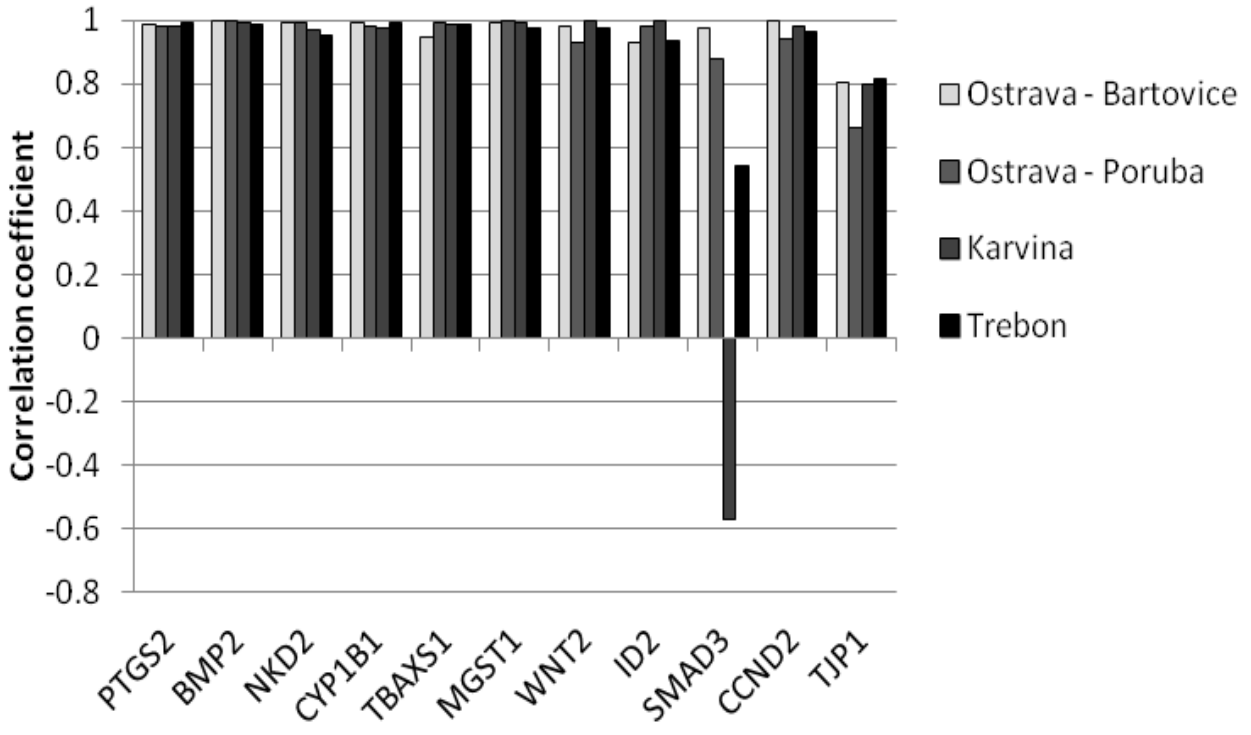


Figure 7 Correlation of the expression of selected transcripts from microarrays and quantitative real-time PCR analysis. The mean correlation across all the transcripts and localities was r = 0.91, (Pearson correlation coefficient).

central role in the induction of drug-metabolizing enzymes. AhR can also interact with other pathways suggesting that this activity is important in the toxicity of exogenous compounds. AhR activation by some of its ligands participates, among others, in pathways involved in the oxidative stress response, cell cycle control and apoptosis [36], cell adhesion and matrix remodeling [37] as well as multiple developmental pathways [19]. Strong involvement of AhR related pathways in the toxic response of EOMs within this study are supported by recent findings on the mechanisms of toxicity induced by an organic extract of the urban dust standard reference material, SRM1649a [14], also suggesting a crucial role for AhR and PAHs as key AhR activators. Another group of upregulated genes within the metabolism of xenobiotics by cytochrome P450 pathway, are glutathione S-transferases *MGST1* and *GSTM5*, known to be involved in conjugation of reduced glutathione to a wide number hydrophobic electrophile metabolites [38,39], and *GSTO1*, a glutathione-dependent thiol transferase and dehydroascorbate reductase [40].

In this study, *CYP1B1* upregulation was also a major factor in deregulation of the second most important deregulated KEGG pathway - Steroid hormone biosynthesis (SHB), which is known to be a target for endocrine-disrupting chemicals [41]. Similar to the Metabolism of xenobiotics by cytochrome P450, this pathway was deregulated in all 4 localities. In addition to *CYP1B1*, which is known to hydroxylate estrogens [42], aldo-keto reductase (*AKR1C2*) and some hydroxysteroid β-dehydrogenases 1 (*HSD17B2, HSD11B1, HSD17B8*) significantly contributed to deregulation of the SHB pathway. The results of the detailed chemical analysis of the EOMs, including the analysis of the dioxin toxicity of the fractionated crude extracts, strongly suggests that PAHs (abundant components in all EOMs) and not persistent organic pollutants (chlorinated dibenzo-p-dioxins, dibenzofuranes or biphenyls) are mainly responsible for SHB deregulation and dioxin-like toxicity. These findings are in accordance with our previous study [14], in which the activation of AhR and AhR-mediated gene expression and cellular nongenotoxic events prevailed the genotoxic and apoptotic processes. Several mechanisms on the modulation of estrogen and androgen signaling by chemicals directly or indirectly through cross-talk between AhR and steroid hormone receptors have been discussed [43,44]. Here we found a possible correlation between the modulations of enzymes of steroidogenesis and oxidative steroid metabolism and the AhR activation.

The crucial role of AhR in the toxic effects of the EOM components is further underlined by the analysis of the third KEGG pathway - Wnt signaling, which was

significantly deregulated by all 4 extracts in this study. Wnt signaling proteins are required for basic developmental processes in many different organs. A recent genomic analysis revealed functional cross-talk between AhR and the well-established Wnt/β-catenin signal transduction pathway [45,46]. *NKD2*, as an upregulated gene mostly contributing to the deregulation of Wnt signaling in all EOM-treated cells, is known as a cell autonomous antagonist of the canonical Wnt signaling pathway [47]. Accordingly, Wnt target genes, *WNT2* and *CCND2* were significantly downregulated in lung fibroblasts exposed to all 4 extracts (Figure 6).

The next KEGG pathway deregulated by EOMs from all 4 localities was the Transforming growth factor-β(*TGF-β*) signaling, which includes structurally related cytokines regulating a wide spectrum of cellular functions such as cell growth and proliferation, apoptosis, differentiation and migration via receptors type I and II [48]. In our study, the upregulation of bone morphogenic protein type-2 (*BMP2*), antagonist of TGF-β involved in osteogenesis, cell differentiation, growth and invasivity, is primarily responsible for TGF-β signaling deregulation [49]. Simultaneously, downregulation of *SMAD3*, effector of TGF-β signaling, and DNA-binding inhibitors 1 or 2 (*ID1, ID2*), transcription factors which negatively regulate cell differentiation [50], was observed in this study. Again, suppression of TGF-β signaling by activated AhR has been reported [51], pointing out the key role of AhR activation in lung fibroblasts exposed to complex airborne mixtures.

The ATP binding cassette (ABC transporters), as a fifth pathway deregulated by all extracts in this study, includes a huge number of various transmembrane proteins capable of active transport of various compounds through the cell membrane. In humans, there are 49 known ABC transporters, which are classified into eight families [52]. In our study, the most significant deregulation was observed for family C (C4 and C5), known to facilitate transport of bile salt and steroid conjugates, ion transport and toxin excretion activity, and family B (B6 and B9), used mostly for transport of peptides [53]. Recently, AhR-dependent upregulation of *ABCC4* was reported [54]. In eukaryotes including humans, ABC transporters serve as pumps that extrude toxins from the cell. Some ABC proteins are known to be involved in translation and DNA repair processes. It was obvious that exposure of HEL cells to complex mixtures of organic compounds bound to PM2.5 induced deregulation of many ABC transporters as a defending reaction of cells to this exposure and that AhR induction may play a significant role in their upregulation. Similar primary transcription response was reported in mouse lung fibroblasts exposed to TCDD for 4 h [55]. TCDD-induced significant upregulation of *CYP1B1, PTGS2, BMP2, ABCC4* and deregulation of other genes belonging to Metabolism of xenobiotics, ABC transporters, TGF-beta and Wnt signaling pathways in mouse lung fibroblasts suggests a significant AhR-dependent gene expression in the HEL cells.

The last pathway deregulated in all 4 localities was Glycerolipid metabolism, which includes the chemical reactions and pathways involving glycerolipids, the lipid with a glycerol backbone. Diacylglycerol and phosphatidic acid are key lipid intermediates of glycerolipid biosynthesis; diacylglycerol is a key lipid signaling molecule involved in activation of protein kinases and cell survival and proliferation. The deregulation of this pathway is caused mainly by upregulation of aldo-keto reductase 1B1 (*AKR1B1*) catalyzing the NADPH-dependent reduction of a wide variety of carbonyl-containing compounds to their corresponding alcohols with a broad range of catalytic efficiencies [56]. It is very likely that carbonyl compounds are the components of all 4 EOMs. Importantly, genes responsible for sphingolipid metabolism were also significantly deregulated; these results suggest effects on sphingolipid signaling molecules which regulate cell survival, proliferation and apoptosis.

There were many other deregulated pathways detected in this study (Table 5), but these were not found in all localities, e.g. Glutathione metabolism pathway was deregulated after the treatment with the extracts from sampling sites Ostrava-Bartovice, Ostrava-Poruba and Karvina but not after the exposure to extract from Trebon,

Table 5 Sequences of primers used in quantitative RT-PCR

Symbol	RefSeq ID	Oligonucleotide	
CYP1B1	NM_000104.2	sense	CACTGGAAACCGCACCTC
		antisense	AGCACCGACAGGAGTAGC
MGST1	NM_145792.1	sense	CACCTGAATGACCTTGAAAATATTATT
		antisense	TCCGTGCTCCGACAAATAGT
NKD2	NM_033120.2	sense	GGAAGGTCACCAGGGAGGA
		antisense	TTCACACGGAGGGTCTTGC
BMP2	NM_001200.2	sense	GGGCATCCTCTCCACAAAAG
		antisense	CCACGTCACTGAAGTCCAC
SMAD3	NM_005902.3	sense	GGCTGCTCTCCAATGTCAAC
		antisense	ACCTCCCCTCCGATGTAGTA
TBXAS1	NM_001061.2	sense	ATCTTCCTCATCGCTGGCTAT
		antisense	CCTTAAAAACGTCTACCTCTCCA
CCND2	NM_001759.2	sense	TGGGACAATGGGTGGTGAA
		antisense	GCAAAGCTGGCTCTTGAGAA
PTGS2	NM_000963.1	sense	CAAATCATCAACACTGCCTCAAT
		antisense	TCTGGATCTGGAACACTGAATG
TJP1	NM_175610.2	sense	AAACAAGCCAGCAGAGACC
		antisense	CGCAGACGATGTTCATAGTTTC
WNT2	NM_003391.1	sense	CAAGAACGCTGACTGGACAA
		antisense	CCCCAGAAAGAACCCAAAGG
ID2	NM_002166.4	sense	CGATGAGCCTGCTATACAACA
		antisense	AGGTCCAAGATGTAGTCGATGA

the only agricultural area. The genes in the KEGG cluster of Glutathione metabolism belong to the phase II biotransformation of xenobiotics and oxidative stress defense. The genes of arachidonic metabolism are involved in proinflammatory responses (*PTGS2, TBXAS1*) and protection against oxidative stress (*GPX3*). Interestingly, the p53 signaling pathway was deregulated exclusively after the treatment by EOM from Ostrava-Bartovice, the most polluted industrial locality. It was found no induction of p53 target genes suggesting possible suppresive role of activated AhR [20]. Genes mostly involved in the deregulation of the Gap junction pathway belong rather to microtubule functions, mitosis and tight junction, they are not so relevant for gap junctions. In conclusion, major deregulated KEGG pathways are related to various cancer promoting processes.

Limitations of this study

The major limitation of this study was that the comparison of the various localities, in terms of the gene expression profiles, should only be regarded as qualitative since equal EOM doses used for all localities (10-60 μg EOM/ml) did not reflect different EOM content per m^3 of the sampled air. In contrast to some toxicity markers such as stable DNA adduct formation [1], gene expression data cannot be normalized to EOM/m^3. Therefore, to make a quantitative comparison of the effect of organic compounds bound to PM2.5 on gene expression profiles, the EOM doses used for cell treatment should take into account the differences in EOM/m^3. Such a study is in progress. On the other hand, using of equal EOM doses allowed us to reveal similar gene expression profiles and affected KEGG pathways.

For technical reasons, it was impossible to sample PM2.5 simultaneously in all 4 localities, which is another limitation of the study. This fact may partially explain why the agricultural locality of Trebon sampled in November and December (period of frequent winter inversions) exhibited such a high number of deregulated transcripts compared to Ostrava-Poruba and Karvina, industrial locations sampled in March and April, respectively. The effect of the winter inversions on particulate matter and PAH air pollution is well known [57].

Conclusion

To our knowledge, this is the first study dealing with differential gene expression in the context of real complex mixtures of air pollutants at the level of the whole genome in human lung fibroblasts. The study identified KEGG pathways deregulated by real complex mixtures of air pollutants collected in areas differing in the extent and sources of air pollution and the key role of activation of AhR was found. The results of this study may be used for future more detailed mechanistic studies focused on the role of individual affected pathways and genes.

Materials and methods

Reagents

All chemical standards were purchased from Promochem (Wesel, Germany), Dr. Ehrenstorfer (Augsburg, Germany) or Midwest Research Institute (Kansas City, MO, USA); solvents were from Merck (Darmstadt, Germany) and chromatographic consumables from Sigma-Aldrich (Prague, Czech Republic). The other compounds and materials used were of the highest purity available suitable for organic trace analysis. DMSO was purchased from Merck, Darmstadt, Germany. The sources of other specific chemicals and kits are indicated below.

PM2.5 collection, sampling sites and EOM extraction

Particulate matter < 2.5 μm (PM2.5) was collected by a HiVol 3000 air sampler (model ECO-HVS3000, Ecotech, Australia) on Pallflex filters T60A20 (20 × 25 cm) in four localities of the Czech Republic differing in the extent and major sources of air pollution: Ostrava-Bartovice (heavily polluted industrial area), Ostrava-Poruba (high level of traffic), Karvina (industrial area) and Trebon (rural area with some houses equipped with local brown coal heating) as described by Topinka et al. [31]. Briefly, sampling was conducted for 24 h each day for 30-35 days in the winter season of 2008/2009. Each filter was extracted by 60 ml of dichloromethane and 3 ml of cyclohexane for 3 hours. The extracts (EOMs) from all filters with PM2.5 samples were pooled and aliquots were used for the detailed chemical analysis and the cell treatment. The extraction of PM2.5 was performed in the laboratories of the certified company ALS Czech Republic, Prague (EN ISO CSN IEC 17025). For the *in vitro* experiments, EOM samples were evaporated to dryness under a stream of nitrogen and the residue redissolved in dimethylsulfoxide (DMSO). The stock solution of each EOM sample contained 50 mg of EOM/ml DMSO. Samples were kept in the freezer at -80°C until analysis.

Sample handling for chemical analysis

Extracts of air $PM_{2.5}$ samples were used for fractionation into four fractions using low-pressure silica gel column chromatography. Fractionation was performed to facilitate the chemical analysis of complex mixtures of polar and nonpolar contaminants of the air samples. An aliquot of the sample extract in dichloromethane was evaporated just to dryness; the residues was redissolved in 0.5 ml of hexane and applied to the top of the open silica gel column. The silica gel (Silica gel 60, particle size 0.063-0.2 mm, Merck, Darmstadt, Germany) was activated for 1 hour at 200°C prior to its use. A column

with the dimensions 250 × 10 mm was dry-packed with 10 g of activated silica gel and washed with 30 ml of hexane prior to the application of the sample. Fractionation was done by gradual elution with 20 ml of hexane to obtain an aliphatic fraction (this fraction was used for alkanes, terpanes and steranes analysis), followed by 20 ml of hexane/dichloromethane (1:1, v/v) (fraction including parent aromatic and POPs compounds), 20 ml of dichloromethane (fraction with slightly-polar compounds such as nitrated derivatives of PAHs) and finally by 30 ml of methanol (polar compounds represented by oxygenated derivatives of PAHs, heterocyclic PAHs with one atom of nitrogen, esters of phthalic acid and sterols). Aliquots of these fractions were redissolved in the required volume of acetonitrile for HPLC/DAD, LC/MS-MS and in 2,2,4-trimethylpentane for GC/MS analysis.

HPLC, LC/MS-MS and GC/MS analysis

The HPLC system consisted of a Waters 717 plus autosampler, a Waters 600 E multisolvent delivery system, a Waters 474 scanning fluorescence detector and a Waters 996 photodiode array detector (Waters, Milford, MA, USA). A 150 × 3 mm Supelcosil LC-PAH column with particle diameter 5 μm (Supelco, Bellefonte, PA, USA) was used for the separation of parent PAHs with molecular weights (MW) ranging from 178 to 326 g/mol. A gradient with water, methanol, acetonitrile and tetrahydrofuran was applied to separate the analytes: 0-55 min. 40-0% water, 30% acetonitrile and 30-70% methanol, 55-72 min. 30-100% acetonitrile and 70-0% methanol, 72-100 min. 100-72% acetonitrile and 0-28% tetrahydrofuran. The flow rate of the mobile phase was 0.6 ml/min., the column temperature was set at 35°C.

The LC/MS-MS analysis of parent PAHs (178-326 MW) and nitrated and oxygenated derivatives of PAHs was performed on a TripleQuad 6410 triple quadrupole mass spectrometer (Agilent, Santa Clara, CA, USA) equipped with an electrospray ion source (ESI), an Agilent 1200 Binary Pump System with an autosampler and a MassHunter software system. The ionization of the analytes was performed in the positive ion mode. The analyte classes were separated in a reverse-phase mode using a Supelcosil LC-PAH HPLC column (150 mm × 3 mm, 5 μm - Supelco, Bellefonte, PA, USA).

Other classes of contaminants included hydrocarbon markers, parent PAHs (128-278 MW), alkylated, oxidized and nitrated derivatives of PAHs and compounds with one heterocyclic atom in the ring (PANHs, PASHs) were determined by GC/MS. GC separation was done in a fused silica capillary column (SLB-5 ms: 30 m × 0.20 mm × 0.20 μm - Sigma-Aldrich, Prague, Czech Republic) with helium as the carrier gas. A Saturn 2100 T ion trap mass spectrometer (Varian, Walnut Creek, CA, USA), which operated in electron ionization and selected ion storage modes at an electron ionization energy of 70 eV, was used for the identification and quantification of the analytes under study.

Cell cultures and cytotoxicity

Human embryonic lung diploid fibroblasts (HEL 12469a, ECACC, UK) were grown in minimal essential medium E-MEM supplemented with 10% FBS, 2 mM glutamine, 1% non-essential amino acids, 0.2% sodium bicarbonate, 50 U/ml penicillin and 50 μg/ml streptomycin. The cells were cultivated in plastic cell culture dishes (21 cm^2) at 37°C in 5% CO_2. After reaching 90% confluency, the medium was replaced with fresh medium supplemented with 1% FBS. EOM samples were diluted by DMSO and added to the medium at the test concentrations: 10, 30 and 60 μg/ml. The cells were treated for 24 h. Each concentration was tested in triplicate including control cell cultures incubated with DMSO only. The harvested cells were washed three times in PBS and the final concentration of DMSO did not exceed 0.1% of the total incubation volume. The cytotoxicity of the extracts in HEL cells was tested by the LDH-Cytotoxicity Assay Kit (Bio Vision, catalogue #K311-400) at concentrations 10, 30, 60 and 100 μg EOM/ml. Significant cytotoxicity was observed at the highest EOM concentration of 100 μg/ml for extracts from Ostrava-Bartovice and Trebon (66% and 17%, respectively). Therefore, three subtoxic EOM concentrations between 10 and 60 μg/ml were used.

RNA isolation and quality control

Total RNA from lysed HEL cells was obtained using NucleoSpin RNA II (Macherey-Nagel GmbH & Co.KG, Düren, Germany) according to the manufacturer's instructions. RNA concentration was quantified with a Nanodrop ND-1000 Spectrophotometer (Thermo Fisher Scientific, Waltham, MA, USA). The integrity of RNA was assessed using an Agilent 2100 Bioanalyzer (Agilent Technologies Inc., Santa Clara, CA, USA). All samples had an RNA Integrity Number (RIN) above 9. Isolated RNA was stored at -80°C until processing.

Gene expression profiling and data analysis

Illumina Human-HT12 v3 Expression BeadChips (Illumina, San Diego, CA, USA) were used to generate expression profiles. Biotinylated cRNAs were prepared from 200 ng of total RNA using the Illumina TotalPrep RNA Amplification Kit (Ambion, Austin, TX, USA). Next, 750 ng of biotinylated cRNA targets was hybridized to the beadchips. The steps of hybridization and the subsequent washing, staining and drying of the beadchips were processed according to standard instructions from Illumina. The

hybridized beadchips were then scanned on the Illumina BeadArray Reader and bead level data were summarized by Illumina BeadStudio Software v2.

Quantitative RT-PCR verification

Two thousands ng RNA from each sample was used for cDNA synthesis using the High Fidelity cDNA synthesis Kit (Roche, Manheim, Germany). The original protocol was modified by using 2.5 μM oligo(dT) and 10 μM random hexamers for priming in a 20 μl reaction volume. cDNA synthesis was run according to the following conditions: 30 min at 55°C and 5 min at 85°C. Quantitative PCR measurements were performed using the 7900 HT Fast Real-Time PCR System (Applied Biosystems, Carlsbad, CA, USA). Each qPCR reaction was carried out in a final volume of 14 μl containing 3.5 μl of diluted cDNA, 2.8 μl of water and 7 μl of master mix (Primerdesign, Southampton, UK). To determine the level of each target gene, 0.7 μl of a specifically designed assay (PerfectProbe, Primerdesign) was added to the reaction mixture (list of primers in Table 5). Cycling conditions were: 10 min at 95°C followed by 40 cycles of amplification (15 s at 95°C, 30 s at 50°C and 15 s at 72°C). Raw data were analyzed with SDS Relative Quantification Software version 2.3 (Applied Biosystems, USA) to assign the baseline and threshold for Ct determination. The sequences of primers used in quantitative RT-PCR are shown in Table 5.

Statistical analysis

Gene expression levels were compared with control HEL cell cultures treated with DMSO only. Bead summary data were imported into R statistical environment http://www.r-project.org and normalized using the quantile method in the Lumi package [58]. Only probes with a detection *P*-value < 0.01 in more than 50% of arrays were included for further analyses. Differential gene expression was analyzed in the Limma package using the moderated *t*-statistic. A linear model was fitted for each gene given a series of arrays using lmFit function [59]. Multiple testing correction was performed using the Benjamini & Hochberg method. A Venn diagram was prepared according to Oliveros [60].

Goeman's global test [61] and the KEGG database were applied to identify deregulated biological pathways and deregulated genes within these pathways. The procedure of Holm for control of the family-wise error rate was applied [62]. The Jonckheere - Terpstra monotonicity test [63,64] was used to analyze the dose response of expression of selected genes.

Ct values of real-time PCR data were analyzed using GenEx software version 5.2.7 (MultiD Analyses AB, Goteborg, Sweden). The expression levels of the target genes were normalized to the expression levels of the reference genes *GAPDH* and *SDHA*. Reference genes were selected according to the stability of gene expression during experimental conditions using the geNorm reference gene selection kit (Primerdesign).

Additional material

Additional file 1: Supplementary table with the list of chemical compounds identified and quantified in EOMs from various localities.

Additional file 2: Complete list of significantly deregulated genes in HEL cells treated with 10, 30, and 60 μg/ml of organic extracts from PM2.5 collected in Ostrava-Bartovice, Ostrava-Poruba, Karvina, and Trebon. Each excel sheet contains list of deregulated transcripts detected from the comparison of gene expression profile of cells treated with an appropriate EOM (Table 1) and cells treated with DMSO. Transcripts with adjusted p-value > 0.05 and average expression < 4 were filtred out.

List of abbreviations

ABC: ATP binding cassettes; AhR: aryl hydrocarbon receptor; B[a]P: benzo[a]pyrene; DCM: dichloromethane; EOM: extractable organic matter; HPLC: high performance liquid chromatography; PAHs: polycyclic aromatic hydrocarbons; PM2.5: particulate matter < 2.5 μm; KEGG: Kyoto Encyclopedia of Genes and Genomes; LDH: lactate dehydrogenase; qPCR: quantitative real time PCR; RIN: RNA integrity number, TGF-β: transforming growth factor beta; SHB: steroid hormone biosynthesis.

Acknowledgements

We would like to thank V. Švecová from the Institute of Experimental Medicine for excellent assistance during the particulate matter sampling campaign. The study was supported by the Czech Ministry of Education (CZ: MSMT 2B08005), Grant Agency of the Czech Republic (CZ: GACR P503/11/0142) and by the Academy of Science of the Czech Republic (CZ: AV CR AV0Z50390512). The work of Jiří Kléma was funded by the Czech Ministry of Education in the framework of the research program, Transdisciplinary Research in the Area of Biomedical Engineering II (MSM 6840770012).

Author details

[1]Department of Genetic Ecotoxicology, Institute of Experimental Medicine, Academy of Sciences of the Czech Republic, 142 20 Prague 4, Czech Republic. [2]Department of Biochemistry, Faculty of Science, Charles University, Albertov 2030, 128 40 Prague 2, Czech Republic. [3]Czech Technical University in Prague, Prague 2, Czech Republic. [4]Veterinary Research Institute, Brno, Czech Republic.

Authors' contributions

HL carried out gene expression analysis on Illumina microarrays and qPCR verification. She also substantially contributed to the description of results and their discussion and interpretation. KU carried out cell preparations and treatment, cytotoxicity analysis and gene expression on Illumina microarrays. JK was responsible for biostatistical evaluation of data. MM substantially contributed to the data interpretation. RJS performed overall text revision, and he contributed to the Discussion. MC was responsible for detailed chemical analysis of extracts from particulate matter. JT was responsible for PM2.5 sampling, EOM extraction, chemical analysis and the overall preparation of the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Received: 10 October 2011 Accepted: 12 January 2012
Published: 12 January 2012

References

1. Lewtas J: **Air pollution combustion emissions: characterisation of causative agents and mechanisms associated with cancer, reproductive and cardiovascular effects.** *Mutat Res* 2007, **636**:95-133.
2. Harrison RM, Smith DJT, Kibble AJ: **What is responsible for the carcinogenicity of PM2.5?** *Occup Environ Med* 2004, **61**:799-805.
3. Claxton LD, Woodall GM Jr: **A review of the mutagenicity and rodent carcinogenicity of ambient air.** *Mutat Res* 2007, **636**:36-94.
4. Topinka J, Hovorka J, Milcova A, Schmuczerova J, Krouzek J, Rossner P Jr, Sram RJ: **Acellular assay to assess genotoxicity of complex mixtures of organic pollutants bound on size segregated aerosol. Part I: DNA adducts.** *Toxocol Lett* 2010, **198**:304-311.
5. Marvin CH, Hewitt LM: **Analytical methods in bioassay-directed investigations of mutagenicity of air particulate material.** *Mutat Res* 2007, **636**:4-35.
6. Karlsson HL, Nygren J, Moller L: **Genotoxicity of airborne particulate matter: the role of cell-particle interaction and of substances with adduct-forming and oxidizing capacity.** *Mutat Res* 2004, **565**:1-10.
7. Steenhof M, Gosens I, Strak M, Godri K, Hoek G, Cassee FR, Mudway IS, Kelly FJ, Harrison RM, Lebret E, Brunekreef B, Janssen NAH, Pieters HHP: **In vitro toxicity of particulate matter (PM) collected at different sites in t5he Netherlands is associated with PM composition, size fraction and oxidative potential - the RAPTES project.** *Particle and Fibre Toxicology* 2011, **8**:26.
8. Topinka J, Schwarz LR, Wiebel FJ, Cerna M, Wolff T: **Genotoxicity of urban air pollutants in the Czech Republic Part II. DNA adduct formation in mammalian cells by extractable organic matter.** *Mutat Res* 2000, **469**:83-93.
9. Binkova B, Sram RJ: **The genotoxic effect of carcinogenic PAHs, their artificial and environmental mixtures (EOM) on human diploid lung fibroblasts.** *Mutat Res* 2004, **547**:109-121.
10. Ghio AJ, Stonehuerner J, Dailey LA, Carter JD: **Metals associated with both the water soluble and insoluble fractions of an ambient air pollution particles catalyze an oxidative stress.** *Inhal Toxicol* 1999, **11**:37-49.
11. Prahalad AK, Inmon J, Dailey LA, Madden MC, Ghio AJ, Gallagher JE: **Air pollution particles mediated oxidative DNA base damage in a cell free system and in human airway epithelial cells in relation to particulate metal content and bioreactivity.** *Chem Res Toxicol* 2001, **14**:879-887.
12. Knaapen AM, Shi T, Borm PJ, Schins RP: **Soluble metals as well as insoluble particle fraction are involved in cellular DNA damage induced by particulate matter.** *Mol Cell Biochem* 2002, **234**:317-326.
13. Upham BL, Bláha L, Babica P, Park JS, Sovadinová I, Pudrith C, Rummel AM, Weis LM, Sai K, Tithof PK, Guzvić M, Vondráček J, Machala M, Trosko JE: **Tumor promoting properties of a cigarette smoke prevalent polycyclic aromatic hydrocarbon as indicated by the inhibition of gap junctional intercellular communication via phosphatidylcholine-specific phospholipase C.** *Cancer Sci* 2008, **99**:696-705.
14. Andrysík Z, Vondráček J, Marvanová S, Cigánek M, Neča J, Pěnčíková K, Mahadevan B, Topinka J, Baird WM, Kozubík A, Machala M: **Activation of the aryl hydrocarbon receptor is the major toxic mode of action of an organic extract of a reference urban dust particulate matter mixture: The role of polycyclic aromatic hydrocarbons.** *Mutation Res* 2011, **714**:53-62.
15. **IARC Monographs on the Evaluation of Carcinogenic Risks to Humans.** 2009 [http://monographs.iarc.fr/ENG/Classification/index.php].
16. Hemminki K, Dipple A, Shuker DEG, Kadlubar FF, Segerbäck D, Bartsch H: **DNA Adducts: Identification and Biological Significance.** *IARC, Lyon* 1994.
17. Dipple A: **DNA adducts of chemical carcinogens.** *Carcinogenesis* 1995, **16**:437-441.
18. Khalili H, Zhang FJ, Harvey RG, Dipple A: **Mutagenicity of benzo[a]pyrene-deoxyadenosine adducts in a sequence context derived from the p53 gene.** *Mutat Res* 2000, **465**:39-44.
19. Puga A, Tomlinson CR, Xia Y: **Ah receptor signals cross-talk with multiple developmental pathways.** *Biochem Pharmacol* 2005, **69**:199-207.
20. Chopra M, Schrenk D: **Dioxin toxicity, aryl hydrocarbon receptor signaling, and apoptosis-persistent pollutants affect programmed cell death.** *Crit Rev Toxicol* 2011, **41**:292-320.
21. Ferecatu I, Borot M-C, Bossard C, Leroux M, Boggetto N, Marano F, Baeza-Squiban A, Andreau K: **Polycyclic aromatic hydrocarbon components contribute to the mitochondria-antiapoptotic effect of fine particulate matter on human bronchial epithelial cells via the aryl hydrocarbon receptor.** *Particle and Fibre Toxicology* 2010, **7**:8-18.
22. Bláha L, Kapplová P, Vondráček J, Upham B, Machala M: **Inhibition of gap-junctional intercellular communication by environmentally occurring polycyclic aromatic hydrocarbons.** *Toxicol Sci* 2002, **65**:43-51.
23. Mahadevan B, Parsons H, Musafia T, Sharma AK, Amin S, *et al*: **Effect of artificial mixtures of environmental polycyclic aromatic hydrocarbons present in coal tar, urban dust, and diesel exhaust particulates on MCF-7 cells in culture.** *Environ Mol Mutagen* 2004, **44**:99-107.
24. Sen B, Mahadevan B, DeMarini D: **Transcriptional responses of the complex mixtures - A review.** *Mutat Res* 2007, **636**:144-177.
25. Mahadevan B, Keshava C, Musafia-Jeknic T, Pecaj A, Weston A, Baird WM: **Altered gene expression patterns in MCF-7 cells induced by the urban dust particulate complex mixture standard reference material 1649a.** *Cancer Res* 2005, **65**:1251-1258.
26. Aung HH, Lame MW, Gohil K, He G, Denison M, Rutledge JC, Wilson DW: **Comparative gene responses to collected ambient particles in vitro: endothelial responses.** *Physiol Genomics* 2011, **43**:917-929.
27. Binkova B, Giguere Y, Rossner P, Dostal M, Sram RJ: **The effect of dibenzo [al]pyrene on human diploid fibroblasts; the induction of DNA adducts, expression of p53 and $p21^{WAF1}$ proteins and cell cycle distribution.** *Mutat Res* 2000, **471**:57-70.
28. Ravindra K, Sokhi R, van Grieken R: **Atmospheric polycyclic aromatic hydrocarbons: source attribution, emission factors and regulation.** *Atmosph Environ* 2008, **42**:2895-2921.
29. Yang Y, Guo P, Zhang Li D, Zhao L, Mu D: **Seasonal variation, sources and gas/particle partitioning of polycyclic aromatic hydrocarbons in Guangzhou, China.** *Science Tot Environ* 2010, **408**:2492-2500.
30. Gupta S, Kumar K, Srivastava A, Srivastava VK: **Size distribution and source apportionment of polycyclic aromatic hydrocarbons (PAHs) in aerosol particle samples from the atmospheric environment of Delhi, India.** *Science of the Total Environment* 2011, **409**:4674-4680.
31. Topinka J, Rossner P Jr, Milcova A, Schmuczerova J, Svecova V, Sram RJ: **DNA adducts and oxidative DNA damage induced by organic extracts from PM2.5 in an acellular assay.** *Toxicol Lett* 2011, **202**:186-192.
32. Guengerich FP, Parikh A, Yun CH, Kim D, Nakamura K, Notley LM, Gillam EM: **What makes P450s work? Searches for answers with known and new P450s.** *Drug Metab Rev* 2000, **32**:267-281.
33. Nebert DW, Russell DW: **Clinical importance of the cytochromes P450.** *Lancet* 2002, **360**:1155-62.
34. Port JL, Yamaguchi K, Du B, De Lorenzo M, Chang M, Heerdt PM, Kopelovich R, Marcus CB, Altorki NK, Subbaramaiah K, Dannenberg A: **Tobacco smoke induces CYP1B1 in the aerodigestive tract.** *Carcinogenesis* 2004, **25**:2275-2281.
35. Tsuchiya Y, Nakajima M, Kyo S, Kanaya T, Inoue M, Yokoi T: **Human CYP1B1 is regulated by estradiol via estrogen receptor.** *Cancer Res* 2004, **64**:3119-3125.
36. Nebert DW, Roe AL, Dieter MZ, Solis WA, Yang Y, Dalton TP: **Role of the aromatic hydrocarbon receptor and [Ah] gene battery in the oxidative stress response, cell cycle control, and apoptosis.** *Biochem Pharmacol* 2000, **59**:65-85.
37. Kung T, Murphy KA, White LA: **The aryl hydrocarbon receptor (AhR) pathway as a regulatory pathway for cell adhesion and matrix metabolism.** *Biochem Pharmacol* 2009, **77**:536-546.
38. Kelner MJ, Stokely MN, Stoval NE, Montoya MA: **Structural organization of the human microsomal glutathione S-transferase gene (GST12).** *Genomics* 1996, **36**:100-103.
39. Xu S, Wang Y, Roe B, Pearson WR: **Characterization of the human class Mu glutathione S-transferase gene cluster and the GSTM1 deletion.** *J Biol Chem* 1998, **6**:3517-3527.
40. Whitbread AK, Masoumi A, Tetlow N, Schmuck E, Coggan M, Board PG: **Characterization of the omega class of glutathione transferases.** *Methods Enzymol* 2005, **401**:78-99.
41. Sanderson JT: **The steroid hormone biosynthesis pathway as a target for endocrine-disrupting chemicals.** *Toxicol Sci* 2006, **94**:3-21.
42. Tsuchiya Y, Nakajima M, Yokoi T: **Cytochrome P450-mediated metabolism of estrogens and its regulation in human.** *Cancer Lett* 2005, **227**:115-124.
43. Ohtake F, Baba A, Fujii-Kuriyama Y, Kato S: **Intrinsic AhR function underlies cross-talk of dioxins with sex hormone signalings.** *Biochem Biophysis Res Commun* 2008, **370**:541-546.

44. Shanle EK, Xu W: **Endocrine disrupting chemicals targeting estrogen receptor signaling: identification and mechanisms of action.** *Chem Res Toxicol* 2011, **24**:6-19.
45. Mathew LR, Simonich MT, Tanguay RL: **AhR-dependent misregulation of Wnt signalling disrupts tissue regeneration.** *Biochem Pharmacol* 2009, **77**:498-507.
46. Procházková J, Kabátková M, Bryja V, Umannová L, Bernatík O, Kozubík A, Machala M, Vondráček J: **The Interplay of the Aryl Hydrocarbon Receptor and β-Catenin Alters Both AhR-Dependent Transcription and Wnt/β-Catenin Signaling in Liver Progenitors.** *Toxicol Sci* 2011, **122**:349-360.
47. Katoh M: **Molecular cloning, gene structure, and expression analyses of NKD1 and NKD2.** *Int J Oncol* 2001, **19**:963-969.
48. Shi Y, Massague J: **Mechanisms of TGF-beta signaling from cell membrane to the nucleus.** *Cell* **113**:685-700.
49. McLean K, Gong Y, Choi Y, Deng N, Yang K, Bai S, Cabrera L, Keller E, McCauley L, Cho KR, Buckanovich RJ: **Human ovarian carcinoma-associated mesenchymal stem cells regulate cancer stem cells and tumorigenesis via altered BMP production.** *J Clin Invest* 2011, **121**:3206-3219.
50. Langlands K, Yin X, Anand G, Prochownik EV: **Differential interactions of Id proteins with basic-helix-loop-helix transcription factors.** *J Biol Chem* 1997, **272**:19785-19793.
51. Gomez-Duran A, Carvajal-Gonzalez JM, Mulero-Navarro S, Santiago-Josefat B, Puga A, Fernandez-Salguero PM: **Fitting a xenobiotic receptor into cell homeostasis: how the dioxin receptor interacts with TGFbeta signaling.** *Biochem Pharmacol* 2009, **15**:700-712.
52. Vasiliou V, Vasiliou K, Nebert DW: **Human ATP-binding cassette (ABC) transporter family.** *Hum Genomics* 2009, **3**:281-290.
53. Davidson AL, Dassa E, Orelle C, Chen J: **Structure, function, and evolution of bacterial ATP-binding cassette systems.** *Microbiol Mol Biol Rev* 2008, **72**:317-364.
54. Xu S, Weerachayaphorn J, Cai SY, Soroka CJ, Boyer JL: **Aryl hydrocarbon receptor and NF-E2-related factor 2 are key regulators of human MRP4 expression.** *Am J Physiol Gastrointest Liver Physiol* 2010, **99**:G126-G135.
55. Henry EC, Welle SL, Gasiewicz TA: **TCDD and a Putative Endogenous AhR Ligand, ITE, Elicit the Same Immediate Changes in Gene Expression in Mouse Lung Fibroblasts.** *Toxicol Sci* 2010, **114**:90-100.
56. Nishimura C, Matsuura Y, Kohai Y, Ahera T, Carper D, Morjana N, Lyons C, Flynn TG: **Cloning and expression of human aldose-reductase.** *J Biol Chem* 1990, **265**:9788-9792.
57. Sevastyanova O, Novakova Z, Hanzalova K, Binkova B, Sram RJ, Topinka J: **Temporal variation in the genotoxic potential of urban air particulate matter.** *Mutat Res* 2008, **649**:179-186.
58. Du P, Kibbe W, Lin SM: **Lumi: a pipeline for processing Illumina microarray.** *Bioinformatics* 2008, **24**:1547-1548.
59. Smyth GK: **Linear models and empirical bayes methods for assessing differential expression in microarray experiments.** *Stat Appl Genet Mol Biol* 2004, **3, Article 3.**
60. Oliveros JC: **VENNY. An interactive tool for comparing lists with Venn Diagrams.**[http://bioinfogp.cnb.csic.es/tools/venny/index.html].
61. Goeman JJ, van de Geer SA, de Kort F, van Houwelingen JC: **A global test for groups of genes: testing association with a clinical outcome.** *Bioinformatics* 2004, **20**:93-99.
62. Holm S: **A simple sequentially rejective multiple test procedure.** *Scand J Stat* 1979, **6**:65-70.
63. Jonckheere AR: **A test of significance for the relation between *m* rankings and *k* ranked categories.** *Brit J Stat Psychol* 1954, **7**:93-100.
64. Bewick V, Cheek L, Ball J: **Statistics review 10: Further nonparametric methods.** *Crit Care* 2004, **8**:196-199.

doi:10.1186/1743-8977-9-1
Cite this article as: Líbalová *et al.*: Global gene expression changes in human embryonic lung fibroblasts induced by organic extracts from respirable air particles. *Particle and Fibre Toxicology* 2012 **9**:1.

Chapter XIII
Gene Expression Mining Guided by Background Knowledge

Jiří Kléma
Czech Technical University in Prague, Czech Republic

Filip Železný
Czech Technical University in Prague, Czech Republic

Igor Trajkovski
Jožef Stefan Institute, Slovenia

Filip Karel
Czech Technical University in Prague, Czech Republic

Bruno Crémilleux
Université de Caen, France

Jakub Tolar
University of Minnesota, USA

ABSTRACT

This chapter points out the role of genomic background knowledge in gene expression data mining. The authors demonstrate its application in several tasks such as relational descriptive analysis, constraint-based knowledge discovery, feature selection and construction or quantitative association rule mining. The chapter also accentuates diversity of background knowledge. In genomics, it can be stored in formats such as free texts, ontologies, pathways, links among biological entities, and many others. The authors hope that understanding of automated integration of heterogeneous data sources helps researchers to reach compact and transparent as well as biologically valid and plausible results of their gene-expression data analysis.

INTRODUCTION

High-throughput technologies like microrarrays or SAGE are at the center of a revolution in biotechnology, allowing researchers to simultaneously monitor the expression of tens of thousands of genes. However, gene-expression data analysis represents a difficult task as the data usually show an inconveniently low ratio of samples (biological situations) against variables (genes). Datasets are often noisy and they contain a great part of variables irrelevant in the context under consideration. Independent of the platform and the analysis methods used, the result of a gene-expression experiment should be driven, annotated or at least verified against genomic background knowledge (BK).

As an example, let us consider a list of genes found to be differentially expressed in different types of tissues. A common challenge faced by the researchers is to translate such gene lists into a better understanding of the underlying biological phenomena. Manual or semi-automated analysis of large-scale biological data sets typically requires biological experts with vast knowledge of many genes, to decipher the known biology accounting for genes with correlated experimental patterns. The goal is to identify the relevant "functions", or the global cellular activities, at work in the experiment. Experts routinely scan gene expression clusters to see if any of the clusters are explained by a known biological function. Efficient interpretation of this data is challenging because the number and diversity of genes exceed the ability of any single researcher to track the complex relationships hidden in the data sets. However, much of the information relevant to the data is contained in publicly available gene ontologies and annotations. Including this additional data as a direct knowledge source for any algorithmic strategy may greatly facilitate the analysis.

This chapter gives a summary of our recent experience in mining of transcriptomic data. The chapter accentuates the potential of genomic background knowledge stored in various formats such as free texts, ontologies, pathways, links among biological entities, etc. It shows the ways in which heterogeneous background knowledge can be preprocessed and subsequently applied to improve various learning and data mining techniques. In particular, the chapter demonstrates an application of background knowledge in the following tasks:

- Relational descriptive analysis
- Constraint-based knowledge discovery
- Feature selection and construction (and its impact on classification accuracy)
- Quantitative association rule mining

The chapter starts with an overview of genomic datasets and accompanying background knowledge analyzed in the text. Section on relational descriptive analysis presents a method to identify groups of differentially expressed genes that have functional similarity in background knowledge. Section on genomic classification focuses on methods helping to increase accuracy and understandability of classifiers by incorporation of background knowledge into the learning process. Section on constraint-based knowledge discovery presents and discusses several background knowledge representations enabling effective mining of meaningful over-expression patterns representing intrinsic associations among genes and biological situations. Section on association rule mining briefly introduces a quantitative algorithm suitable for real-valued expression data and demonstrates utilization of background knowledge for pruning of its output ruleset. Conclusion summarizes the chapter content and gives our future plans in further integration of the presented techniques.

GENE-EXPRESSION DATASETS AND BACKGROUND KNOWLEDGE

The following paragraphs give a brief overview of information resources used in the chapter. The primary role of background knowledge is to functionally describe individual genes and to quantify their similarity.

Gene-Expression (Transcriptome) Datasets

The process of transcribing a gene's DNA sequence into the RNA that serves as a template for protein production is known as gene expression. A gene's expression level indicates an approximate number of copies of the gene's RNA produced in a cell. This is considered to be correlated with the amount of corresponding protein made.

Expression chips (DNA chips, microarrays), manufactured using technologies derived from computer-chip production, can now measure the expression of thousands of genes simultaneously, under different conditions. A typical gene expression data set is a matrix, with each column representing a gene and each row representing a class labeled sample, e.g. a patient diagnosed having a specific sort of cancer. The value at each position in the matrix represents the expression of a gene for the given sample (see Figure 1). The particular problem used as an example in this chapter aims at distinguishing between acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) (Golub, 1999). The gene expression profiles were obtained by the Affymetrix HU6800 microarray chip, containing probes for 7129 genes, the data contains 72 class-labeled samples of expression vectors. 47 samples belong to the ALL class (65%) as opposed to 25 samples annotated as AML (35%).

SAGE (Serial Analysis of Gene Expression) is another technique that aims to measure the expression levels of genes in a cell population (Velculescu, 1995). It is performed by sequencing tags (short sequences of 14 to 21 base pairs (bps) which are specific of each mRNA). A SAGE library is a list of transcripts expressed at one given time point in one given biological situation. Both the identity (assessed through a tag-to-gene complex process, (Keime, 2004)) and the amount of each transcript is recorded. SAGE, as a data source, has been largely under-exploited as of today, in spite of its important advantage over microarrays. In fact, SAGE can produce datasets that can be directly compared between libraries without the need for external normalization. The human transcriptome can be seen as a set of libraries that would ideally be collected in each biologically relevant situation in the human body. This is clearly out of reach at the moment, and we deal in the present work with 207 different situations ranging from embryonic stem cells to foreskin primary fibroblast cells. Unambiguous tags (those that enable unequivocal gene identification) were selected leaving a set of 11082 tags/genes. A 207x11082 gene expression matrix was built.

Figure 1. The outcome of a microarray or SAGE experiment

<table>
<tr><td></td><td>gene 1</td><td>gene 2</td><td>...</td><td>gene n</td><td>target</td></tr>
<tr><td>sample/situation 1</td><td colspan="4" rowspan="4">Values of gene expression (binary, symbolic, integer or real)
Sample expression signatures in rows,
gene expression profiles in columns</td><td>T_1</td></tr>
<tr><td>sample/situation 2</td><td>T_2</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>sample/situation m</td><td>T_m</td></tr>
</table>

The biological situations embody various tissues (brain, prostate, breast, kidney or heart) stricken by various possible diseases (mainly cancer, but also HIV and healthy tissues). As the main observed disorder is carcinoma, a target binary attribute Cancer was introduced by the domain expert. The class value is 0 for all the healthy tissues and also the tissues suffering by other diseases than cancer (77 situations in total, 37.2%). It is equal to 1 for all the cancerous tissues (130 situations, 62.8%). The dataset was also binarized to encode the over-expression of each gene using the MidRange method described in (Becquet, 2002). For each gene it takes its highest value (max), the lowest value (min), and calculates the mid-range as (max-min)/2. Values above the threshold are given a boolean value of 1; all others are given a value of 0.

Background Knowledge

In this chapter, the term genomic background knowledge refers to any information that is not available in a gene-expression dataset but it is related to the genes or situations contained in this dataset. The richest body of background knowledge is available for genes. Gene datatabases such as Entrez Gene (NCBI website: http://www.ncbi.nlm.nih.gov/) offer a large scale of gene data – general information including a short textual summary of gene function, cellular location, bibliography, interactions and further links with other genes, memberships in pathways, referential sequences and many other pieces of information. Having a list of genes (i.e. colums in Figure 1), the information about all of the genes from the list can be collected automatically via services such as Entrez Utils (NCBI website: http://www.ncbi.nlm.nih.gov/). Similarly, annotations of the biological samples (i.e. rows in Figure 1) contained in the gene-expression dataset are available. In the simplest case, there is at least a brief description of the aim of the experiment where the sample was used.

Two forms of external knowledge require special attention during data pre-processing. These are freetexts and gene ontologies (GOs). We use them in two principal ways. The first way of utilization extracts all the relevant keywords for each gene, the main purpose is to **annotate**. In the second way we aim to **link** the genes. We introduce a quantitative notion of gene similarity that later on contributes to the cost-efficient reduction of computational costs across various learning and data mining algorithms. In the area of freetexts we have been inspired mainly by (Chaussabel, 2002; Glenisson, 2003). Both of them deal with the term-frequency vector representation which is a simple however prevailing representation of texts. This representation allows for an annotation of a gene group as well as a straightforward definition of gene similarity. In the area of gene ontologies we mostly rely on (Martin, 2004), the gene similarity results from the genes' positions in the molecular functional, biological process or cellular component ontology.

However, alternative sources can also be used, e.g., (Sevon, 2006) suggests an approach to discover links between entities in biological databases. Information extracted from available databases is represented as a graph, where vertices correspond to entities and edges represent annotated relationships among entities. A link is manifested as a path or a sub-graph connecting the corresponding vertices. Link goodness is based on edge reliability, relevance and rarity. Obviously, the graph itself or a corresponding similarity matrix based on the link goodness can serve as an external knowledge source.

Free Texts and Their Preprocessing

To access the gene annotation data for every gene or tag considered, probe identifiers (in the case microarrays) or Reference Sequence (RefSeq) identifiers (for SAGE) were translated into Entrez Gene

Identifiers (Entrez Ids) using the web-tool MatchMiner (http://discover.nci.nih.gov/ matchminer/). The mapping approached 1 to 1 relationship. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the EntrezGene database and sequentially parsed (Klema, 2006). Non-trivial textual records were obtained for the majority of the total amount of unique ids. The gene textual annotations were converted into the vector space model. A single gene corresponds to a single vector, whose components correspond to the frequency of a single vocabulary term in the text. This representation is often referred to as bag-of-words (Salton, 1988). The particular vocabulary consisted of all stemmed terms (Porter stemmer, http://www.tartarus.org/~martin/ PorterStemmer/) that appear in 5 different gene records at least. The most frequent terms were manually checked and insufficiently precise terms (such as gene, protein, human etc.) were removed. The resulting vocabulary consisted of 17122 (ALL/AML), respectively 19373 terms (SAGE). The similarity between genes was defined as the cosine of the angle between the corresponding term-frequency inverse-document-frequency (TFIDF) (Salton, 1988) vectors. The TFIDF representation statistically considers how important a term is to a gene record.

A similarity matrix *s* for all the genes was generated (see Figure 2). Each field of the triangular matrix $s_{ij} \in \langle 0,1 \rangle$ gives a similarity measure between the genes *i* and *j*. The underlying idea is that a high value of two vectors' cosine (which means a low angle among two vectors and thus a similar occurrence of the terms) indicates a semantic connection between the corresponding gene records and consequently their presumable connection. This model is known to generate false positive relations (as it does not consider context) as well as false negative relations (mainly because of synonyms). Despite this inaccuracy, bag-of-words format corresponds to the commonly used representation of text documents. It enables efficient execution of algorithms such as clustering, learning, classification or visualization, often with surprisingly faithful results (Scheffer, 2002).

Gene Ontology

One of the most important tools for the representation and processing of information about gene products and functions is the Gene Ontology (GO). It provides a controlled vocabulary of terms for the description of cellular components, molecular functions, and biological processes. The ontology also identifies those pairs of terms where one **is a** special case of the other. Similarly, term pairs are identified where

Figure 2. Gene similarity matrix – the similarity values lie in a range from 0 (total mismatch between gene descriptions) to 1 (pertfect match), n/a value suggests that at least one of the gene tuple has no knowledge attached

	gene 1	gene 2	gene 3	gene 4	...	gene n
gene 1	1	0.05	n/a	n/a	...	0.63
gene 2		1	0.01	0.33	...	0.12
gene 3			1	n/a	...	n/a
gene 4				1	...	n/a
...	...	...	...	...	...	...
gene n						1

one term refers to a **part of** the other. Formally this knowledge is reflected by the binary relations "**is a"** and "**part of**".

For each gene we extracted its ontological annotation, that is, the set of ontology terms relevant to the gene. This information was transformed into the gene's background knowledge encoded in relational logic in the form of Prolog facts. For example, part of the knowledge for particular gene SRC, whose EntrezId is 6714, is as follows:

```
function(6714,'ATP binding').
function(6714,'receptor activity').
process(6714,'signal complex formation').
process(6714,'protein kinase cascade').
component(6714,'integral to membrane').
...
```

Next, using GO, in the gene's background knowledge we also included the gene's generalized annotations in the sense of the "**is a**" relation described above. For example, if one gene is functionally annotated as: "zinc ion binding", in the background knowledge we also included its more general functional annotations such as e.g. transition metal ion binding or metal ion binding.

The genes can also be functionally related on the basis of their GO terms. Intuitively, the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related. (Martin, 2004) defines a distance based on the Czekanowski-Dice formula, the methodology is implemented within the GOProxy tool of GOToolBox http://crfb.univ-mrs.fr/GOToolBox/). A similarity matrix of the same structure as shown in Figure 2 can be generated.

Gene Interactions

The similarity matrix described in the previous paragraphs is one specific way to represent putative gene interactions. Besides, public databases also offer the information about pairs of genes for which there is traced experimental evidence of mutual interaction. In this case we use a crisp declarative representation of the interaction, in the form of a Prolog fact. The following example represents an interaction between gene SRC (EntrezId 6714) and genes ADRB3 (EntrezId 155) and E2F4 (EntrezId 1874):

```
interaction(6714,155).
interaction(6714,1874).
```

RELATIONAL DESCRIPTIVE ANALYSIS

This section presents a method to identify groups of differentially expressed genes that have functional similarity in background knowledge formally represented by gene annotation terms from the gene ontology (Trajkovski, 2006). The input to the algorithm is a multidimensional numerical data set, representing the expression of the genes under different conditions (that define the classes of examples), and an ontology used for producing background knowledge about these genes. The output is a set of

gene groups whose expression is significantly different for one class compared to the other classes. The distinguishing property of the method is that the discovered gene groups are described in a rich, yet human-readable language. Specifically, each such group is defined in terms of a logical conjunction of features, that each member of the group possesses. The features are again logical statements that describe gene properties using gene ontology terms and interactions with other genes.

Medical experts are usually not satisfied with a separate description of every important gene, but want to know the processes that are controlled by these genes. The presented algorithm enables to find these processes and the cellular components where they are "executed", indicating the genes from the pre-selected list of differentially expressed genes which are included in these processes.

These goals are achieved by using the methodology of Relational Subgroup Discovery (RSD) (Lavrac, 2002). RSD is able to induce sets of rules characterizing the differentially expressed genes in terms of functional knowledge extracted from the gene ontology and information about gene interactions.

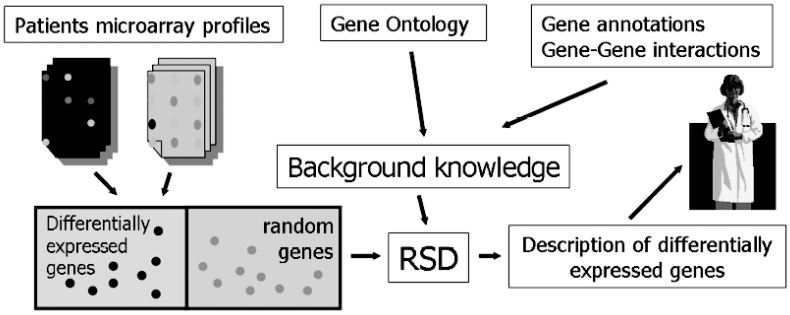
Fundamental Idea

The fundamental idea of learning relational descriptions of differentially expressed gene groups is outlined in Figure 3 (Trajkovski 2008). First, a set of differentially expressed genes, $G_C(c)$, is constructed for every class $c \in C$ (e.g. types of cancer). These sets can be constructed in several ways. For example: $G_C(c)$ can be the set of k ($k > 0$) most correlated genes with class c, for instance computed by Pearson's correlation. $G_C(c)$ can also be the set of best k single gene predictors, using the recall values from a microarray/SAGE experiment (absent/present/marginal) as the expression value of the gene. These predictors can acquire the form such as:

If $gene_i$ = present Then class = c

In our experiments, $G_C(c)$ was constructed using a modified version of the t-test statistics. The modification lies in an additional condition ensuring that each selected gene has at least twofold difference in its average expression for the given class with respect to the rest of the samples. The second step aims at improving the interpretability of G_C. Informally, we do this by identifying subgroups of genes in $G_C(c)$ (for each $c \in C$) which can be summarized in a compact way. Put differently, for each $c_i \in C$ we search for

Figure 3. An outline of the process of gene-expression data analysis using RSD



(© 2008 IEEE Transactions on Systems, Man, and Cybernetics: Part C. Used with permission.)

compact descriptions of gene subgroups with expression strongly correlating (positively or negatively) with c_i and weakly with all $c_j \in C$; $j \neq i$.

Searching for these groups of genes, together with their description, is defined as a supervised machine learning task. We refer to it as the secondary mining task, as it aims to mine from the outputs of the primary learning process in which differentially expressed genes are searched. This secondary task is, in a way, orthogonal to the primary discovery process in that the original attributes (genes) now become training examples, each of which has a class label "differentially expressed" and "not differentially expressed". Using the gene ontology information, gene annotation and gene interaction data, we produce background knowledge for differentially expressed genes on one hand, and randomly chosen genes on the other hand. The background knowledge is represented in the form of Prolog facts. Next, the RSD algorithm finds characteristic descriptions of the differentially expressed genes. Finally, the discovered descriptions can be straightforwardly interpreted and exploited by medical experts.

Relational Subgroup Discovery

The RSD algorithm proceeds in two steps. First, it constructs a set of relational features in the form of first-order logic atom conjunctions. The entire set of features is then viewed as an attribute set, where an attribute has the value true for a gene (example) if the gene has the feature corresponding to the attribute. As a result, by means of relational feature construction we achieve the conversion of relational data into attribute-value descriptions. In the second step, interesting gene subgroups are searched, such that each subgroup is represented as a conjunction of selected features. The subgroup discovery algorithm employed in this second step is an adaptation of the popular propositional rule learning algorithm CN2 (Clark, 1989).

The feature construction component of RSD aims at generating a set of relational features in the form of relational logic atom conjunctions. For example, the informal feature *"gene g interacts with another gene whose functions include protein binding"* has the relational logic form:

interaction(g,B), function(B,'protein binding')

where upper cases denote variables, and a comma between two logical literals denotes a conjunction. The user specifies mode declarations which syntactically constrain the resulting set of constructed features and restrict the feature search space. Furthermore, the maximum length of a feature (number of contained literals) is declared. RSD proceeds to produce an exhaustive set of features satisfying the declarations. Technically, this is implemented as an exhaustive depth-first backtrack search in the space of all feature descriptions, equipped with certain pruning mechanisms. Finally, to evaluate the truth value of each feature for each example for generating the attribute-value representation of the relational data, the first-order logic resolution procedure is used, provided by a standard Prolog language interpreter.

Subgroup discovery aims at finding population subgroups that are statistically "most interesting", e.g., are as large as possible and have the most unusual statistical characteristics with respect to the target class. To discover interesting subgroups of genes defined in terms of the constructed features, RSD follows a strategy stemming from the popular rule learner CN2. See (Zelezny, 2006) for details on this procedure.

Experiments

In ALL, RSD has identified a group of 23 genes, described as a conjunction of two features: component(G,'nucleus') AND interaction(G,B),process(B,'regulation of transcription, DNA-dependent'). The products of these genes, proteins, are located in the nucleus of the cell, and they interact with genes that are included in the process of regulation of transcription. In AML, RSD has identified several groups of overexpressed genes, located in the membrane, that interact with genes that have 'metal ion transport' as one of their function.

Subtypes of ALL and AML can also be distinguished, in a separate subgroup discovery process where classes are redefined to correspond to the respective disease subtypes. For example, two subgroups were found with unusually high frequency of the BCR (TEL, respectively) subtype of ALL. The natural language description of BCR class derived from the automatically constructed subgroup relational description is the following: *genes coding for proteins located in the integral to membrane cell component, whose functions include receptor activity*. This description indeed appears plausible, since BCR is a classic example of a leukemia driven by spurious expression of a fusion protein expressed as a continuously active kinase protein on the *membrane* of leukemic cells. Similarly, the natural language description for the TEL class is: genes coding for proteins located in the nucleus whose functions include protein binding and whose related processes include transcription. Here again, by contrast to BCR, the TEL leukemia is driven by expression of a protein, which is a transcription factor active in the *nucleus*.

A statistical validation of the proposed methodology for discovering descriptions of differentially expressed gene groups was also carried out. The analysis determined if the high descriptive capacity pertaining to the incorporation of the expressive relational logic language incurs a risk of descriptive overfitting, i.e., a risk of discovering subgroups whose bias toward differential expression is only due to chance. The discrepancy of the quality of discovered subgroups on the training data set on one hand and an independent test set on the other hand was measured. It was done through the standard 10-fold stratified cross-validation regime. The specific qualities measured for each set of subgroups produced for a given class are average precision (PRE), recall (REC) and area under ROC (AUC) values among all subgroups in the subgroup set. In ALL/AML dataset, RSD showed PRE 100(±0)%, REC 16% and AUC 65% in training data and PRE 85(±6)%, REC 13% and AUC 60% in independent testing data. The results demonstrate an acceptable decay from the training to the testing set in terms of both PRE and REC, suggesting that the discovered subgroup descriptions indeed capture the relevant gene properties. In terms of total coverage, in average, RSD covered more then 2/3 of the preselected differentially expressed genes, while 1/3 of the preselected genes were not included in any group. A possible interpretation is that they are not functionally connected with the other genes and their initial selection through the t-test was due to chance. This information can evidently be back-translated into the gene selection procedure and used as a gene selection heuristic.

GENOMIC CLASSIFICATION WITH BACKGROUND KNOWLEDGE

Traditional attribute-value classification searches for a mapping from attribute value tuples, which characterize instances, to a discrete set whose elements correspond to classes. When dealing with a large number of attributes and a small number of instances, the resulting classifier is likely to fit the training

data solely by chance, rather than by capturing genuine underlying trends. Datasets are often noisy and they contain a great part of variables irrelevant in the context of desired classification.

In order to increase the predictive power of the classifier and its understandability, it is advisable to incorporate background knowledge into the learning process. In this section we study and test several simple ways to improve a genomic classifier constructed from gene expression data as well as textual and gene ontology annotations available both for the genes and the biological situations.

Motivation

Decision-tree learners, rule-based classifiers or neural networks are known to often overfit gene expression data, i.e., identify many false connections. A principal means to combat the risk of overfitting is feature selection (FS); a process aiming to filter irrelevant variables (genes) from the dataset prior to the actual construction of a classifier. Families of classifiers are available, that are more tolerant to abundance of irrelevant attributes than the above mentioned traditional methods. Random forests (Breiman, 2001; Diaz-Uriarte, 2006) or support vector machines (Furey, 2000; Lee, 2003) exemplify the most popular ones. Still, feature selection remains helpful in most gene expression classification analyses. Survey studies (such as (Lee, 2005)) stress that the choice of feature selection methods has much effect on the performance of the subsequently applied classification methods.

In the gene expression domain, feature selection corresponds to the task of finding a limited set of genes that still contains most of the information relevant to the biological situations in question. Many gene selection approaches create rankings of gene relevance regardless of any knowledge of the classification algorithm to be used. These approaches are referred to as filter methods. Besides general filter ranking methods (different modifications of the t-test, information gain, mutual information), various specific gene-selection methods were published. The signal-to-noise (S2N) ratio was introduced in (Golub, 1999), significance analysis of microarrays (SAM) appeared in (Tusher, 2001). (Tibshirani, 2002) proposed and tested nearest shrunken centroids (NSC). The wrapper methods can be viewed as gene selection methods which directly employ classifiers. Gene selection is then guided by analyzing the embedded classifier's performance as well as its result (e.g. to detect which variables proved important for classification). Recursive Feature Elimination (RFE) based on absolute magnitude of the hyperplane elements in a support vector machine is discussed in (Guyon, 2002). (Uriarte, 2006) selects genes according to the decrease of the random forest classification accuracy when values of the gene are permuted randomly.

Here we consider feature selection techniques in a different perspective. All of the above-mentioned methods rely on gene expression data itself. No matter whether they apply a single-variate or multivariate selection criteria, they disregard any potential prior knowledge on gene functions and its true or potential interactions with other genes, diseases or other biological entities. Our principal aim is to exploit background knowledge such as literature and ontologies concerning genes or biological situations as a form of evidence of the genes' relevance to the classification task.

The presented framework uses the well-known CN2 (Clark, 1989) rule learning algorithm. In fact, rule-based classification exhibits a particular weakness when it comes to gene expression data classification. This is due to their small resistance to overfitting, as commented above. As such, a rule learning algorithm is a perfect candidate to evaluate the possible assets of background knowledge. Thus, the main goal is not to develop the best possible classifier in terms of absolute accuracy. Rather, we aim to assess the relative gains obtained by integrating prior knowledge. The evaluated gains pertain to classification accuracy, but also to the comprehensibility of the resulting models.

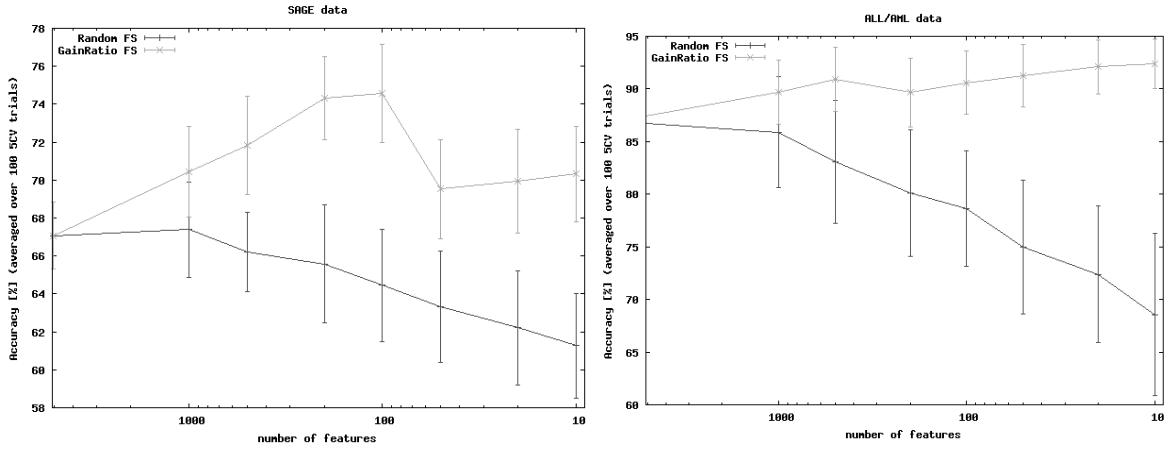
Feature Selection

We first consider the widely accepted dogma that feature selection helps improve classification accuracy and test it in the gene expression domain. A single-variate gain ratio (GR) (Quinlan, 1986) evaluation criterion was used. The criterion is information-based and disregards apriori knowledge. The graphs in Figure 4 show that indeed: 1) FS improves classification accuracy (SAGE dataset – the average accuracy grows from 67.1% for 5052 features to 72.8% for 50 features, ALL/AML dataset – the average accuracy grows from 86.9% for 7129 features to 92.4% for 10 features), 2) informed FS outperforms the random one.

Next, we want to design a mechanism which could guide feature selection using available apriori knowledge. The main idea is to promote the genes whose description contains critical keywords relevant to the classification objective. For example, SAGE classification tries to distinguish among cancerous and non-cancerous tissues. Consequently, the genes that are known to be active in cancerous tissues may prove to be more important than they seem to according to their mutual information with the target (expressed in terms of entropy, gain ratio or mutual information itself). These genes should be promoted into the subset of selected features. Auxiliary experiments proved that there are large gene groups whose mutual information with the target differs only slightly. As a consequence, even an insignificant difference then may decide whether the gene gets selected. To avoid this threshold curse, one may favor multi-criteria gene ranking followed by gene filtering.

The way in which we rank genes with respect to their textual and/or ontological description depends on the amount of information available for biological situations. In the SAGE dataset, each situation contains a brief textual annotation. The frequent words from these annotations serve to create a list of relevant keywords. In the ALL/AML dataset, there are descriptions of the individual classes and the list of keywords is made of the words that characterize these classes. In order to calculate gene importance, the list of keywords is matched with the bag-of-words that characterizes the individual genes. A gene is rated higher if its description contains a higher proportion of situation keywords. Let us show the following simple example:

Figure 4. Gain ratio - development of classification accuracy with decreasing number of features/ genes



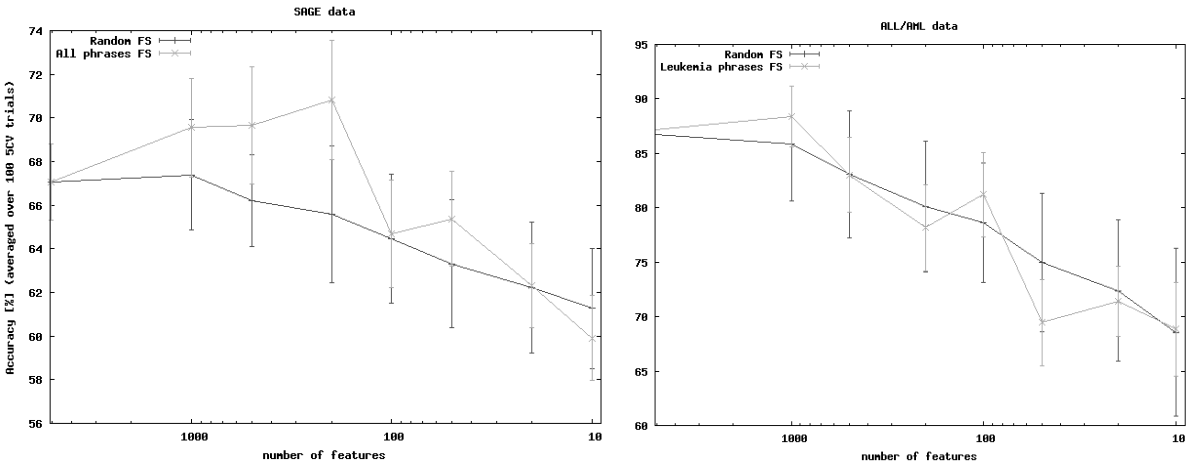
Keywords (characterize the domain): carcinoma, cancer, glioblastoma
Bag of words (characterize the gene): bioactive, cancer, framework, glioblastoma
gene1: 1, 3, 4, 2, gene2: 0, 0, 2, 0 (the word bioactive appears 3 times in gene1 annotations etc.)
gene1 scores (1+4)/(1+3+4+2)=0.5, gene2 scores 0/2=0

We refer to this process as the apriori-based FS. The graphs in Figure 5 compare the apriori-based FS with the random one. In the SAGE dataset, the list of apriori genes is better than random, although the margin is not as distinct as for the information-based criterion used in Figure 4. In the ALL/AML dataset, the apriori-based genes proved to have similar predictive power as randomly selected genes. A likely explanation for this is that the list of keywords was too short. The gene ranking was too rough to correlate with the real gene importance. A great portion of genes scored 0 as they never co-occur with any keyword.

We next tackle the question whether one can cross-fertilize the information-based and apriori-based FS. Two different FS procedures were implemented – conditioning and combination. Conditioning FS keeps the gain ratio ranking but removes all the genes scoring less than a threshold on the apriori-based ranking scale. When asked for X best genes, it takes the X top genes from the reduced list. Combination FS takes the best genes from top of both the lists. When asked for X best genes it takes X/2 top genes from the gain ratio list and X/2 top genes from the apriori list. The result is shown in Figure 6. In spite of better than random quality of apriori-based FS in SAGE dataset, neither conditioning nor combination outperforms gain ratio. The apriori list seems to bring no additional strong predictors. In the ALL/AML dataset, conditioning gives the best performance. It can be explained by the good informativeness of the set of 1000 top genes from the apriori list, which enriches the original gain-ratio list.

In general, the experiments proved that usability of apriori-based FS strongly depends on the domain and the target of classification. The amount of available keywords and their relevance make the crucial issue.

Figure 5. Apriori-based feature selection - development of classification accuracy with decreasing number of features/genes



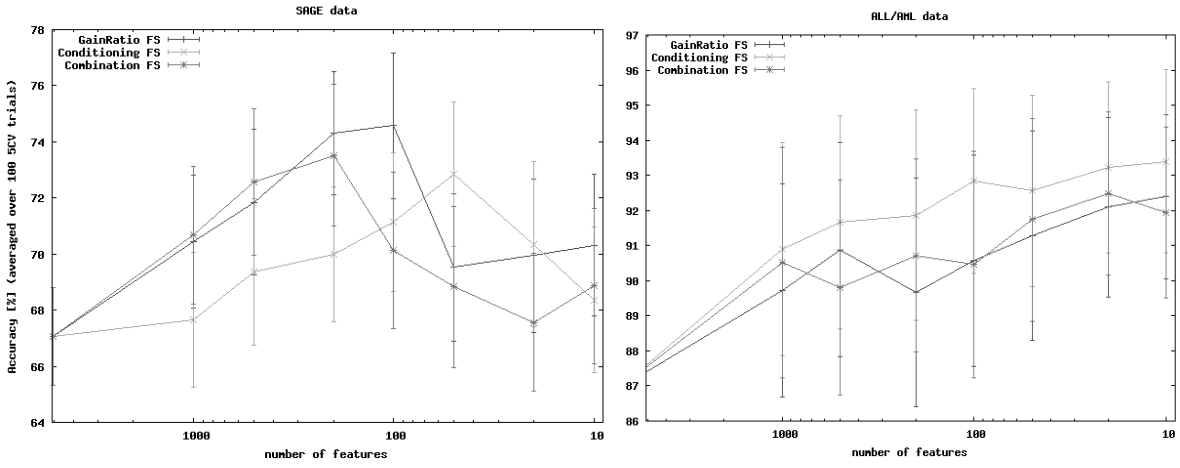
Feature Extraction

The curse of feature space dimensionality can also be overcome or in the least reduced by feature extraction (FE). It is a procedure that transforms the original feature space by building new features from the existing ones. (Hanczar, 2003) proposed a prototype-based feature extraction that consists of two simple steps: 1) identify equivalence classes inside the feature space, 2) extract feature prototypes that represent the classes invented in step 1. In practice, the features are clustered and each cluster is represented by its mean vector – the prototype. The prototypes are used to learn a classifier and to classify new biological situations.

An interesting fact is that equivalence classes can be derived from the gene expression profiles as well as from the known gene functions or any other biologically relevant criteria. The gene similarity matrix based on gene-expression profiles can be combined with the gene-similarity matrices inferred from the background knowledge. Although the prototypes did not prove to increase classification accuracy either in the ALL/AML or the SAGE task, the prototypes can increase understandability of the resulting classifier. The classifier does not treat the individual genes but it reports the equivalence classes whose interpretability is higher as they are likely to contain "similar" genes.

Another idea is to inject background knowledge into the learning algorithm itself. In case of CN2, the algorithm implements a laplacian heuristic that drives rule construction. As mentioned earlier, the algorithm is likely to overfit the data as it searches a large feature space, verifies a large number of simple conditions and randomly finds a rule with a satisfactory heuristic value. Background knowledge can complement the laplacian criteria in the following way: 1) promote short rules containing genes with apriori relevance to the target (a kind of late feature *selection* conditioned by rule length and heuristic value), 2) promote the rules with interacting genes (a kind of late feature *extraction* with the same conditioning). This form of background knowledge injection was implemented and evaluated in (Trna, 2007). The main benefit of this method is the understandability of the resulting classifier.

Figure 6. Combined feature selection - development of classification accuracy with decreasing number of features/genes



CONSTRAINT-BASED KNOWLEDGE DISCOVERY

Current gene co-expression analyses are often based on global approaches such as clustering or bi-clustering. An alternative way is to employ local methods and search for patterns – sets of genes displaying specific expression properties in a set of situations. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate patterns which can hardly be further exploited by a human. A timely application of background knowledge can help to focus on the most plausible patterns only. This section discusses various representations of BK that enables the effective mining and representation of meaningful over-expression patterns representing intrinsic associations among genes and biological situations.

Constraints Inferred from Background Knowledge

Details on knowledge discovery from local patterns are given in another chapter of this book (Cremilleux, 2008). This section focuses on processing, representation and utilization of BK within the constraint-based framework presented ibid. In the domain of constraint-based mining, the constraints should effectively link different datasets and knowledge types. For instance, in the domain of genomics, biologists are interested in constraints both on co-expression groups and common characteristics of the genes and/or biological situations concerned. Such constraints require to tackle transcriptome data (often provided in a transactional format) and external databases. This section provides examples of a declarative language enabling the user to set varied and meaningful constraints defined on transcriptome data, similarity matrices and textual resources.

In our framework, a constraint is a logical conjunction of propositions. A proposition is an arithmetic test such as $C > t$ where t is a number and C denotes a *primitive* or a *compound*. A primitive is one of a small set of predefined simple functions evaluated on the data. Such primitives may further be assembled into compounds.

We illustrate the construction of a constraint through an example. A textual dataset provides a description of genes. Each row contains a list of phrases that characterize the given gene. The phrases can be taken from gene ontology or they can represent frequent relevant keywords from gene bibliography:

Gene 1: ‘metal ion binding’ ‘transcription factor activity’ ‘zinc ion binding’
Gene 2: ‘hydrolase activity’ ‘serine esterase activity’ ‘cytoplasmic membrane-bound vesicle’
...
Gene n: ‘serine-type peptidase activity’ ‘proteolysis’ ‘signal peptide processing’

In reference to the textual data, *regexp(X,RE)* returns the items among X whose phrase matches the regular expression *RE*.

As concerns the similarity matrices, we deal with primitives such as *sumsim(X)* denoting the similarity sum over the set of items X or *insim(X,min,max)* for the number of item pairs whose similarity lies between *min* and *max*. As we may deal with a certain portion of items without any information, there are primitives that distinguish between *zero* similarity and *missing value* of similarity. The primitive *svsim(X)* gives the number of item pairs belonging to X whose mutual similarity is valid and *mvsim(X)* stands for its counterpart, i.e., the missing interactions when one of the items has an empty record

within the given similarity representation. The primitives can make compounds. Among many others, *sumsim(X)/svsim(X)* makes the average similarity, *insim(X,thres,1)/svsim(X)* gives a proportion of the strong interactions (similarity higher than the threshold) within the set of items, *svsim(X)/(svsim(X)+mvsim(X))* can avoid patterns with prevailing items of an unknown function.

Relational and logical operators as well as other built in functions enable to create the final constraint, e.g., $C_1 \geq thres_1$ and $C_2 \neq thres_2$ where C_i stands for an arbitrary compound or primitive. Constraints can also be simultaneously derived from different datasets. Then, the dataset makes another parameter of the primitive. For example, the constraint *length(regexp(X,'*ribosom*', TEXT))>1* returns all the patterns that contain at least 2 items involving "ribosom" in any of their characteristic phrases within the TEXT dataset.

Internal and External Constraints to Reach a Meaningful Limited Pattern Set

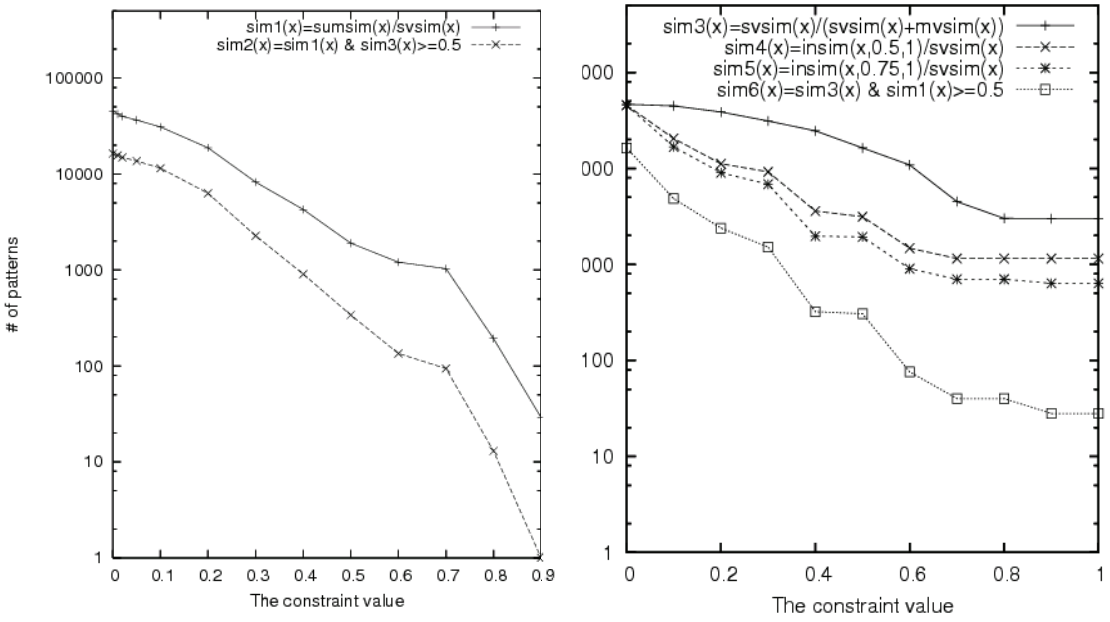
Traditional pattern mining deals with constraints that we refer to as internal. Truth values of such constraints are fully determined by the transcriptome dataset. The most meaningful internal constraints usually are the *area*, i.e. the product of the number of genes in the pattern (gene set), and the frequency of the pattern (number of transactions where the set is contained). This is because the main goal is usually to identify large gene sets that tend to co-occur frequently. For these constraints to apply, one must consider a *binarized* expression dataset enabling to state whether or not a gene is expressed in a given situation. Verifying the area constraints means checking whether the area is larger than a certain threshold.

However, the area constraint is not a panacea for distinction between meaningful patterns and spurious ones, i.e., the patterns occurring randomly. Indeed, the largest area patterns often tend to be trivial, bringing no new knowledge. In SAGE, the increase of the area threshold in order to get a reasonable number of patterns leads to a small but uniform set that is flooded by the ribosomal genes which represent the most frequently over-expressed genes in the dataset. On the other hand, if the area threshold is decreased, the explosion of patterns may occur. It has been experimentally proven that the number of potentially large patterns is so high that they cannot be effectively surveyed by a human expert.

The described deficiency may be healed by augmenting internal constraints by further constraints, called *external*. An external constraint is one whose truth value is determined exclusive of the transcriptome dataset. Such constraints are for example *interestingness or expressiveness*, i.e., the future interpretability by a biologist. The interesting patterns are those exhibiting a general characteristic common for the genes and/or samples concerned (or at least their sub-sets). The more internal functional links in the pattern the more interesting the pattern.

Selectivity of selected external constraints in SAGE dataset is shown in Figure 7. The constraints capture the amount of similarity in given patterns through the measurement of the similarity of all gene pairs within that given pattern as well as they can avoid patterns with prevailing tags of an unknown function. The pruning starts with 46671 patterns that are larger than 3 genes and more frequent than 5 samples. The graphs depict that if both similarity (sumsim or insim) and existence (svsim) are thresholded, very compact sets of patterns can be reached. (Klema, 2006) gives a demonstration that these sets also gather biologically meaningful patterns.

Figure 7. Pattern pruning by the external constraints - simultaneous application of internal and external constraints helps to arbitrarily reduce the number of patterns while attempting to conserve the potentially interesting ones. The figures show the decreasing number of patterns with increasing threshold of selected external constraints. The effect of six different constraints of various complexity is shown



QUANTITATIVE ASSOCIATION RULE MINING IN GENOMICS USING BACKGROUND KNOWLEDGE

Clustering is one of the most often used methods of genomic data mining. The genes with the most similar profiles are found so that the similarity among genes in one group (cluster) is maximized and similarity among particular groups (clusters) is minimized. While clustering arguably is an elegant approach to provide effective insight into data, it does have drawbacks as well, of which we name three (Becquet, 2002):

1. One gene has to be clustered in one and only one group, although it functions in numerous physiological pathways.
2. No relationship can be inferred between the different members of a group. That is, a gene and its target genes will be co-clustered, but the type of relationship cannot be rendered explicit by the algorithm.
3. Most clustering algorithms will make comparisons between the gene expression patterns in all the conditions examined. They will therefore miss a gene grouping that only arises in a subset of cells or conditions.

Admittedly, drawback 1 is tackled by soft-clustering and drawback 2 is tackled by conceptual clustering. We are not aware of a clustering algorithm void of all the three deficiencies.

Association rule (AR) mining can overcome these drawbacks, however transcriptomic data represent a difficult mining context for association rules. First, the data are high-dimensional (typically contain several thousands of attributes), which asks for an algorithm scalable in the number of attributes. Second, expression values are typically quantitative variables. This variable type further increases computational demands and moreover may result in an output with a prohibitive number of redundant rules. Third, the data are often noisy which may also cause a large number of rules of little significance. In this section we discuss the above-mentioned bottlenecks and present results of mining association rules using an alternative approach to quantitative association rule mining. We also demonstrate a way in which background genomic knowledge can be used to prune the search space and reduce the amount of derived rules.

Related Work

One of the first thorough studies of AR mining on genomic data sets was provided in (Becquet, 2002). To validate the general feasibility of association rule mining in this data domain, the authors of (Becquet, 2002) have applied it to a freely available data set of human serial analysis of gene expression (SAGE). The SAGE data was first normalized and binarized as to contain only zeros and ones. These values stand for underexpression and overexpression of a given gene in a given situation, respectively. The authors selected 822 genes in 72 human cell types and generated all frequent and valid rules in the form of 'when gene a and gene b are overexpressed within a situation, then often gene c is over expressed too'.

To avoid this discretization step, authors in (Georgii, 2005) investigate the use of *quantitative association rules*, i.e., association rules that operate directly on numeric data and can represent the cumulative effects of variables. Quantitative association rules have the following form:

If the weighted sum of some variables is greater than a threshold, then, with high probability, a different weighted sum of variables is greater than second threshold.

An example of such rule can be:

$0.99 \times \text{gene1} - 0.11 \times \text{gene2} > 0.062 \rightarrow 1.00 \times \text{gene3} > -0.032.$

This approach naturally overcomes the discretization problem; on the other hand it is quite hard to understand the meaning of the rule. This algorithm does not exhaustively enumerate all valid and strong association rules present in the data, it uses an optimization approach.

An analysis of a microarray data-set is presented in (Carmona-Saez, 2006). The authors bring external biological knowledge to the AR mining by setting a specific language bias In particular, only gene ontology terms are allowed to appear in the antecedent part of the rule. Annotated gene expression data sets can thus be mined for rules such as:

cell cycle → [+]condition1, [+]condition2, [+]condition3, [−]condition6

which means that a significant number of the genes annotated as 'cell cycle' are over-expressed in condition 1, 2 and 3 and under-expressed in condition 6, where the conditions here correspond to time interval <T1..T7>. A proviso for this method is, of course, that ontology annotations are available for all genes in question.

Time Complexity of Association Rule Mining

Time complexity is a serious issue in association rule mining, as it is an exponential function of sample dimensionality. To get a glimpse of the problem size, consider a binarized gene-expression dataset. Here, the number of possible itemsets is $2^{1000} \approx 10^{300}$ Although algorithms such as APRIORI use effective pruning techniques to dramatically reduce the search space traversed, the time complexity bound remains exponential.

With quantitative association rules, things get even worse. Consider the discretization into three bins, when each gene takes three values: 1 – gene is underexpressed, 2 – gene is averagely expressed, 3 – gene is overexpressed. Number of possible conditions (itemsets) grows to $5^{1000} \approx 10^{700}$, because now there are **five** possibilities for gene's value {1; 2; 3; 1..2; 2..3}. Any complete search-based algorithm becomes unfeasible even if it is completed by pruning or puts restrictions on the number of attributes on the left-hand and right-hand side (LHS, RHS). Clearly, the strong restrictions mentioned above have to be complemented by other instruments.

Background Knowledge Experiments in Association Rule Mining

In order to increase noise robustness, focus and speed up the search, it is vital to have a mechanism to exploit BK during AR generation. In the following we employ BK in the form of a similarity matrix as defined earlier. In particular, the similarity matrix describes how likely the genes are functionally related based on the GO terms they share. The experiments are carried out in the frame of the SAGE dataset.

BK is employed in pruning. The pruning takes a following form: generate a rule only if the similarity of the genes contained in the rule is above some defined threshold'. Similarly to constraint-based learning, this condition reduces the search space and helps to speed up the algorithm. It also provides us with results, which could be better semantically explained and/or annotated.

The QAR mining algorithm presented in (Karel, 2006) was used for experiments on SAGE dataset. The QAR algorithm uses a modified procedure of rule generation – it constructs compound conditions using simple mathematical operations. Then it identifies areas of increased association between LHS and RHS. Finally, rules are extracted from these areas of increased association. The procedure is incomplete as it does not guarantee that all the rules satisfying the input conditions are reported. Although the algorithm differs in principle from traditional AR mining, it outputs association rules in the classical form.

The numbers of rules as well as the numbers of candidate rule verifications were examined during the experiments, since the number of rules quantifies the output we are interested in and the number of verifications determinates time complexity of the algorithm.

The SAGE dataset is sparse – a great portion of gene-expression values equal to zero. The distribution of zeroes among genes is very uneven. So called housekeeping genes are expressed (nearly) in all the tissues; however there is a reasonable amount of genes having zero values in almost all situations.

A total of 306 genes having more than 80% non-zero values were used in the experiment. The raw data were preprocessed and discretized into three bins using K-means discretization.

While the right hand side of rules can take arbitrary forms within the language bias, we do fix it to only refer to the target variable *cancer*, as this variable is of primary interest. Additional restrictions needed to be introduced to keep time complexity in reasonable limits. The maximum number of LHS genes was bounded. The results of experiments are summarized in Table 1.

The theoretical number of verifications is computed without considering a *min_supp* pruning, because it is hard to estimate the reached reduction. Numbers of rules and verifications using background knowledge depend on the BK pruning threshold.

A vector *gene_appearance* was generated for the purpose of overall analysis of the results; the value $gene\_appearance_i$ is equal to the number of corresponding gene appearances in generated rules. Spearman rank correlation coefficient among *gene_appearance* vectors of all results was computed, see Table 2.

As we can see using background knowledge we receive most similar rules. Surprising is negative correlation between results with 2 LHS genes with background knowledge and 3 LHS genes without background knowledge. Background knowledge influences not only number of rules generated but also the character of the rules. Some concrete examples of generated rules can be found in Table 3.

Table 1. The number of rules and verifications for 2 and 3 antecedent genes. The settings were following: 2 gene thresholds: min_supp = 0.3, min_conf = 0.7 and min_lift = 1.3, 3 gene thresholds: min_supp = 0.15, min_conf = 0.8 and min_lift = 1.3

algorithm	number of LHS genes	number of rules	number of verifications
complete search (theoretical)	2	n/a	2 318 000
QAR algorithm (without BK)		530	76 747
QAR algorithm (with BK)		92	12 770
complete search (theoretical)	3	n/a	591 090 000
QAR algorithm (without BK)		7 509	14 921 537
QAR algorithm (with BK)		243	699 444

Table 2. Spearman rank correlation coefficients for vectors describing number of genes' appearances in generated rules

	2-ant with BK	2-ant without BK	3-ant with BK	3-ant without BK
2-ant with BK	1	0.04	0.29	-0.25
2-ant without BK	0.04	1	0.09	0.17
3-ant with BK	0.29	0.09	1	0.26
3-ant without BK	-0.25	0.17	0.26	1

Table 3. Examples of generated association rules. For gene expression levels it holds 1 – underexpressed, 2 – averagely expressed, 3 – overexpressed. Consequent condition stands for binary class cancer (0 – cancer did not occur, 1 – cancer did occur).

nr.	antecedent genes and their values	antecedent genes full name	cons. condition	conf	supp	lift
1	RPL31 = 1..2 NONO = 2..3	ribosomal protein L31 non-POU domain containing, octamer-binding	1	0.83	0.35	1.32
2	NONO = 2..3 FKBP8 = 1	non-POU domain containing, octamer-binding FK506 binding protein 8, 38kDa	1	0.81	0.31	1.29
3	MIF = 1..2 CDC42 = 2..3	macrophage migration inhibitory factor (glycosylation-inhibiting factor) cell division cycle 42 (GTP binding protein, 25kDa)	1	0.79	0.30	1.25
4	PHB2 = 2 PGD = 1 LGALS1 = 1	prohibitin 2 phosphogluconate dehydrogenase lectin, galactoside-binding, soluble, 1 (galectin 1)	1	0.94	0.15	1.50
5	COPA = 1..2 CDC42 = 2..3 NDUFS3 = 2..3	coatomer protein complex, subunit alpha cell division cycle 42 (GTP binding protein, 25kDa) NADH dehydrogenase (ubiquinone) Fe-S protein 3, 30kDa (NADH-coenzyme Q reductase)	1	0.90	0.17	1.43
6	PCBP1 = 2..3 ZYX = 1..1 ATP5B = 1..1	poly(rC) binding protein 1 zyxin ATP synthase, H+ transporting, mitochondrial F1 complex, beta polypeptide	1	0.88	0.18	1.40

Discussion

A heuristic QAR approach reduces the number of verifications and thus time costs. The usage of AR mining is extended beyond boolean data and can be applied on genomic data sets, although the number of attributes in the conditions has still to be restricted. The number of generated rules was also reduced by other means – there were at most two rules for each gene tuple. Consequently, the output is not flooded by quantities of rules containing the same genes having only small changes in their values.

Background knowledge was incorporated into QAR mining. BK provides a principled means to significantly reduce the search space and focus on plausible rules only. In general, the genes with prevalence of 'n/a' values in the similarity matrices are discriminated from the rules when using BK. However, a gene without annotation can still appear in a neighbourhood of 'a strong functional cluster' of other

genes. This occurrence then signifies its possible functional relationship with the given group of genes and it can initiate its early annotation. On the other hand, the genes with extensive relationships to the other genes may increase their occurrence in the rules inferred with BK.

CONCLUSION

The discovery of biologically interpretable knowledge from gene expression data is one of the hottest contemporary genomic challenges. As massive volumes of expression data are being generated, intelligent analysis tools are called for. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate hypotheses which can hardly be further post-processed exploited by a human expert. A timely application of background knowledge available in literature, databases, biological ontologies and other sources, can help to focus on the most plausible candidates only. We illustrated a few particular ways how background knowledge can be exploited for this purpose.

Admittedly, the presented approaches to exploiting background knowledge in gene expression data mining were mutually rather isolated, despite their common reliance on the same sources of external genomic knowledge. Intuition suggests that most effective results could be obtained by their pragmatic combination. For example, gene-gene similarity has so far been computed on the sole basis of gene ontology or textual term occurrences in the respective annotations. This definition admittedly may be overly shallow. Here, the RSD mechanism of constructing non-trivial relational logic features of genes may instead be used for computing similarity: two genes would be deemed functionally similar if they shared a sufficient number of the relational logic features referring to gene functions. The inverse look at the problem yields yet another suggestion for combining the methods. The similarity matrix computed from gene ontology term occurrences can be used as a part of background knowledge which RSD uses to construct features. Technically, a new predicate *similar(A,B)* would be introduced into the feature language, while its semantics for two genes *A* and *B* would be determined in the obvious way from the precomputed similarity matrix. These ideas form grounds for our future explorations.

ACKNOWLEDGMENT

Jiří Kléma, Filip Železný and Filip Karel have been supported by the grant 1ET101210513 ”Relational Machine Learning for Analysis of Biomedical Data” funded by the Czech Academy of Sciences. The Czech-French travels were covered by Czech-French PHC Barrande project “Fusion de données hétérogenes pour la découverte de connaissances en génomique’’. The Czech-USA travels were covered by Czech Ministry of Education through project ME910 “Using Gene Ontologies and Annotations for the Interpretation of Gene Expression Data through Relational Machine Learning Algorithms”. We thank Olivier Gandrillon and his team at the Centre de Génétique Moléculaire et Cellulaire for helpful and stimulating biological discussions and providing us with the SAGE dataset. We are grateful to Arnaud Soulet for providing us with his tool MUSIC applied in constraint-based knowledge discovery.

REFERENCES

Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J. F. & Gandrillon O. (2002). Strong Association Rule Mining for Large Gene Expression Data Analysis: A Case Study on Human SAGE Data. *Genome Biology*, 3(12):531-537.

Breiman, L. (2001) Random Forests. *Machine Learning*, 45(1), 5–32.

Carmona-Saez, P., Chagoyen, M., Rodriguez, A., Trelles, O., Carazo, J. M., & Pascual-Montano, A. (2006). Integrated analysis of gene expression by association rules discovery. BMC *Bioinformatics*, *7*, 54.

Chaussabel, D., & Sher, A. (2002). Mining microarray expression data by literature profiling. *Genome Biology, 3*.

Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, 261–283.

Cremilleux, B., Soulet, A., Klema, J., Hebert, C., & Gandrillon, O. (2009). Discovering Knowledge from Local Patterns in SAGE data. In P. Berka, J. Rauch and D. J. Zighed (Eds.), *Data mining and medical knowledge management: Cases and applications*. Hershey, PA: IGI Global.

Diaz-Uriarte, R., & Alvarez de Andres, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, *7*(3).

Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M. & Haussler, D. (2000). Support vector machine classifcation and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, *16*, 906–914.

Georgii, E., Richter, L., Ruckert, U., & Kramer S. (2005) Analyzing Microarray Data Using Quantitative Association Rules. *Bioinformatics*, 21(Suppl. 2), ii123–ii129.

Glenisson, P., Mathys, J., & Moor, B. D. (2003) Meta-clustering of gene expression data and literature-based information. *SIGKDD Explor. Newsl.* 5(2), 101–112.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., & Lander, E. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 531–537.

Hanczar, B., Courtine, M., Benis, A., Hennegar, C., Clement, C. & Zucker, J. D. (2003). Improving classification of microarray data using prototype-based feature selection. *SIGKDD Explor. Newsl.*, *5*(2), 23--30. ACM, NY, USA.

Karel, F. (2006) Quantitative and ordinal association rules mining (QAR mining). In *Knowledge-Based Intelligent Information and Engineering Systems*, 4251, 195–202. Springer LNAI.

Karel, F., & Klema, J. (2007). Quantitative Association Rule Mining in Genomics Using Apriori Knowledge. In Berendt, B., Svatek, V. Zelezny, F. (eds.), Proc. of The *ECML/PKDD Workshop On Prior Conceptual Knowledge in Machine Learning and Data Mining.* University of Warsaw, Poland, (pp. 53-64).

Keime, C., Damiola, F., Mouchiroud, D., Duret, L. & Gandrillon, O. (2004). Identitag, a relational database for SAGE tag identification and interspecies comparison of SAGE libraries. *BMC Bioinformatics*, *5*(143).

Klema, J., Soulet, A., Cremilleux, B., Blachon, S., & Gandrilon, O. (2006). Mining Plausible Patterns from Genomic Data. *Proceedings of Nineteenth IEEE International Symposium on Computer-Based Medical Systems*, Los Alamitos: IEEE Computer Society Press, 183-188.

Klema, J., Soulet, A., Cremilleux, B., Blachon, S., & Gandrilon, O. (submitted). Constraint-Based Knowledge Discovery from SAGE Data. Submitted to *In Silico Biology.*

Lavrac, N., Zelezny, F., & Flach, P. (2002). RSD: Relational subgroup discovery through first-order feature construction. In *Proceedings of the 12th International Conference on Inductive Logic Programming*, 149–165.

Lee, J. W., Lee, J. B., Park, M., & Song, S. H. (2005). An extensive evaluation of recent classification tools applied to microarray data. *Computation Statistics and Data Analysis*, *48*, 869-885.

Lee, Y., & Lee, Ch. K. (2003) Classification of Multiple Cancer Types by Multicategory Support Vector Machines Using Gene Expression Data. *Bioinformatics*, *19*(9), 1132-1139.

Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D., & Jacq, B. (2004). GOToolBox: functional investigation of gene datasets based on Gene Ontology. *Genome Biology 5*(12), R101.

Quinlan, J.R. (1986) Induction of decision trees. *Machine Learning*, *1*, 81-106.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing Management, 24*(5), 513–523.

Scheffer, T., & Wrobel, S. (2002). Text Classification Beyond the Bag-of-Words Representation. Proceedings of the *International Conference on Machine Learning (ICML) Workshop on Text Learning.*

Sevon, P., Eronen, L., Hintsanen, P., Kulovesi, K., & Toivonen, H. (2006). Link discovery in graphs derived from biological databases. In *3rd International Workshop on Data Integration in the Life Sciences* (DILS'06), Hinxton, UK.

Soulet, A., Klema J., & Cremilleux, B. (2007). Efficient Mining Under Rich Constraints Derived from Various Datasets. In Džeroski, S., Struyf, J. (eds.), *Knowledge Discovery in Inductive Databases*, LNCS,4747, 223-239. Springer Berlin / Heidelberg.

Tibshirani, R., Hastie, T., Narasimhan, B., & Chu, G. (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc. *Natl Acad. Sci*., 99(10), 6567-6572.

Trajkovski, I., Zelezny, F., Lavrac, N., & Tolar, J. (in press). Learning Relational Descriptions of Differentially Expressed Gene Groups. *IEEE Trans. Sys Man Cyb C, spec. issue on Intelligent Computation for Bioinformatics.*

Trajkovski, I., Zelezny, F., Tolar, J., & Lavrac, N. (2006) Relational Subgroup Discovery for Descriptive Analysis of Microarray Data. In *Procs 2nd Int Sympos on Computational Life Science*, Cambridge, UK 9/06. Springer Lecture Notes on Bioinformatics / LNCS.

Trna, M. (2007) *Klasifikace s apriorni znalosti*. CTU Bachelor's Thesis, In Czech.

Tusher, V.G., Tibshirani, R. & Chu G. (2001) Significance analysis of microarrays applied to the ionizing radiation response. Proc. *Natl Acad. Sci*., 98(9). 5116–5121.

Velculescu, V., Zhang, L., Vogelstein, B. & Kinzler, K. (1995). Serial Analysis of Gene Expression. *Science, 270*, 484–7.

Zelezny, F., & Lavrac, N. (2006) Propositionalization-Based Relational Subgroup Discovery with RSD. *Machine Learning, 62*(1-2), 33-63.

KEY TERMS

ALL, AML: Leukemia is a form of cancer that begins in the blood-forming cells of the bone marrow, acute leukemias usually develop suddenly (whereas some chronic varieties may exist for years before they are diagnosed), acute myeloid leukemia (AML) is the most frequently reported form of leukemia in adults while acute lymphoblastic leukemia (ALL) is largely a pediatric disease.

Association Rule: A rule, such as implication or correlation, which relates elements co-occurring within a dataset.

Background Knowledge: Information that is essential to understanding a situation or problem, knowledge acquired through study or experience or instruction that can be used to improve the learning process.

Classifier: A mapping from unlabeled instances (a discrete or continuous feature space X) to discrete classes (a discrete set of labels Y), a decision system which accepts values of some features or characteristics of a situation as an input and produces a discrete label as an output.

Constraint: A restriction that defines the focus of search, it can express allowed feature values or any other user's interest.

DNA, RNA, mRNA: Deoxyribonucleic acid (DNA) is a nucleic acid that contains the genetic instructions used in the development and functioning of all known living organisms, ribonucleic acid (RNA) is transcribed from DNA by enzymes, messenger RNA (mRNA) is the RNA that carries information from DNA to the ribosome sites of protein synthesis (translation) in the cell, the coding sequence of the mRNA determines the amino acid sequence in the protein that is produced.

Functional Genomics: A field of molecular biology that attempts to make use of the vast wealth of data produced by genomic projects to describe gene and protein functions and interactions.

Gene Expression: The process of transcribing a gene's DNA sequence into the RNA that serves as a template for protein production.

Gene Ontology: A controlled vocabulary to describe gene and gene product attributes in any organism.

Knowledge Discovery: The nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

RefSeq: Non-redundant curated data representing current knowledge of a known gene.

Relational Data Mining: Knowledge discovery in databases when the database has information about several types of objects.

SRC: V-src sarcoma (Schmidt-Ruppin A-2) viral oncogene homolog (avian) – a randomly taken gene to illustrate knowledge representation format.